UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

AD-A204 322

1b. RESTRICTIVE MARKINGS	
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) AFOSR-TR- 89-0012
6a. NAME OF PERFORMING ORGANIZATION The Pennsylvania State Univers.	6b. OFFICE SYMBOL (If applicable) Aerospace Eng.
7a. NAME OF MONITORING ORGANIZATION AFOSR/NM	
6c. ADDRESS (City, State, and ZIP Code) 233 Hammond Bldg University Park, PA 16802	7b. ADDRESS (City, State, and ZIP Code) AFOSR/NM Bldg 410 Bolling AFB, DC 20332-6448
8a. NAME OF FUNDING / SPONSORING ORGANIZATION AFOSR	8b. OFFICE SYMBOL (If applicable) NM
9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER AFOSR - 87 - 0121	
8c. ADDRESS (City, State, and ZIP Code) AFOSR/NM Bldg 410 Bolling AFB, DC 20332-6448	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
61102F	2304	A3	

DTIC ELECTE 13 FEB 1989 S E

11. TITLE (Include Security Classification)
Generalized Non-Linear Minimal Residual (GNLMR) Method for Optimal Multistep Iterative Algorit

12. PERSONAL AUTHOR(S) Dr. George S. Dulikravich

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final Report	FROM 870115 TO 881014	881206	58

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A new Distributed Minimal Residual (DMR) method for the acceleration of explicit iterative algorithms for the numerical solution of systems of partial differential equations has been developed by Lee and Dulikravich. The method is based on the idea of allowing each partial differential equation in the system to approach the converged solution at its own optimal speed while at the same time communicating with the rest of the equations in the system. The DMR method belongs to a general class of the extrapolation techniques in which the solution is updated using information from a number of consecutive time steps in such a way that the L_2 norm of future residual is minimized. Unlike in other similar methods, each component of the solution vector is updated using a separate sequence of acceleration factors. The idea of using different acceleration factors for each component of a solution vector is similar to that of dynamic preconditioning. This allows each equation to evolve at its own optimal convergence rate.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclass
22a. NAME OF RESPONSIBLE INDIVIDUAL Lt.Col. David Nelson;	22b. TELEPHONE (Include Area Code) (202) 767-5026 — 22c. OFFICE SYMBOL NM

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

Final Technical Report for Period: 15 January, 1987 - 14 October 1988

AFOSR-TR- 89-0012

GENERALIZED NON-LINEAR MINIMAL RESIDUAL (GNLMR) METHOD FOR OPTIMAL MULTI-STEP ITERATIVE ALGORITHMS

(AFOSR-87-0121)

Prepared by:

Dr. George S. Dulikravich, Associate Professor
Department of Aerospace Engineering
Pennsylvania State University
University Park, PA 16802
(814) 863-0134

Accession For	
NTIS GRA&I	X
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Prepared for:

Lt. Col. Dr. David Nelson
Directorate of Mathematical and Information Sciences
AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
Numerical Mathematics Program, Building 410
Bolling AFB, DC 20332-6448
(202) 767-5026

December 12, 1988

89 2 10 109

LIST OF CONTENTS

I. SUMMARY

GENERALIZED NON-LINEAR MINIMAL RESIDUAL (GNLMR) METHOD FOR OPTIMAL MULTISTEPPING

George S. Dulikravich

Penn State University
Department of Aerospace Engineering
233 Hammond Building
University Park, PA 16802

One of the extrapolation methods for the acceleration of iterative algorithms is the Generalized Non-Linear Minimal Residual (GNLMR) concept. It utilizes a number of intermediate steps when advancing the solution to the next time level. That is, numerical error at the new time level can be expressed as a sum of intermediate corrections where each correction is multiplied by a separate acceleration factor which GNLMR optimizes. The method was originally developed by Kennon and Dulikravich and then successfully generalized and applied by Huang and Dulikravich to a number of problems governed by single non-linear partial differential equations. In addition, Huang has obtained preliminary results where GNLMR was successfully applied to a system of four nonlinear partial differential equations (mass conservation, two components of linear momentum equation, and energy equation) governing unsteady two-dimensional flow of compressible, rotational, inviscid fluid. This system is known as Euler equations of gas dynamics. The basic integration algorithm was an explicit scheme that utilizes Runge-Kutta time-stepping and finite volume formulation for spatial discretization. The algorithm is known as Jameson's scheme and represents the fastest presently available integration method for Euler equations of gas dynamics.

A new Distributed Minimal Residual (DMR) method for the acceleration of explicit iterative algorithms for the numerical solution of systems of partial differential equations has been developed by Lee and Dulikravich. The method is based on the idea of allowing each partial differential equation in the system to approach the converged solution at its own optimal speed while at the same time communicating with the rest of the equations in the system. The DMR method belongs to a general class of the extrapolation techniques in which the solution is updated using information from a number of consecutive time steps in such a way that the L_2 norm of future residual is minimized. Unlike in other similar methods, each component of the solution vector is updated using a separate sequence of acceleration factors. The idea of using different acceleration factors for each component of a solution vector is similar to that of dynamic preconditioning. This allows each equation to evolve at its own optimal convergence rate. Moreover, the acceleration factors are determined from the governing equations so that only a few consecutive solutions are required for a successful application of the DMR method. This acceleration scheme was applied to the system of time-dependent Euler equations of inviscid gasdynamics in conjunction with the finite volume Runge-Kutta explicit time-stepping algorithm. Using DMR without multigridding, between 30% and 70% of the total computational efforts were saved in the subsonic compressible flow calculations. The DMR method seems to be especially suitable for stiff systems of equations and can be applied to other systems of differential equations and other numerical algorithms. Specifically, the DMR method was applied to an artificial compressibility, explicit, Runge-Kutta time stepping algorithm for steady, incompressible, Navier-Stokes equations. A two-dimensional analysis computer code in a generalized curvilinear coordinate system was developed and its accuracy has been compared to known numerical solutions. The algorithm has been successfully accelerated using the DMR method, resulting in 25%-70% reduction in computing time.

II. STATUS OF THE RESEARCH

The objective of the research project was to provide a sound mathematical theory for non-linear iterative acceleration schemes using multiple optimal acceleration factors and to test the method on several non-linear differential systems. Particular emphasis was placed on developing computer programs for accelerating the convergence and enhancing stability of iterative solutions of the non-linear systems of partial differential equations of fluid mechanics.

During the course of this research project, both analytical and software development aspects were addressed. A general theory of optimal acceleration factors for the multi-step iterative solution of systems of non-linear partial differential equations based on the minimal residual concept was developed with the special emphasis on mixed-type systems of partial differential equations. The new methods was tested on a variety of practical examples governed by the compressible two-dimensional inviscid flow equations (Euler equations) and viscous incompressible laminar flow equations (Navier-Stokes equations). Subsonic and transonic flow fields were calculated for geometries including nozzles, airfoil cascades, airfoil in an unbounded domain, and a driven cavity problem.

Two graduate students, Mr. Chung-Yuan Huang and Mr. Stephen R. Kennon, have finished their doctorate degrees in the summer of 1987 at the University of Texas at Austin, while supervised by Professor George S. Dulikravich from Penn State University who continued to serve as an adjunct faculty with the University of Texas. Both Dr. Huang and Dr. Kennon were partially supported by this grant and the preceding grant from the AFOSR/NM with Professor David M. Young as co-principal investigator. This fact was acknowledged in their doctoral dissertations.

Dr. Huang has applied GNLMR to a number of scalar nonlinear partial differential equations and to a system of Euler equations of gasdynamics. He now works as a postdoctoral Research Scientist with Professor J. Tinsley Oden at the University of Texas. Dr. Kennon has developed a number of new ideas for finite elements in gasdynamics including acceleration of iterative algorithms. He now works as an Assistant Professor in the Aerospace Engineering Department of the University of Texas at Arlington.

A new graduate research assistant at Penn State was supported with the grant AFOSR-87-0121. Mr. Seungsoo Lee is a Ph.D. candidate who has developed the DMR method and applied it to the explicit finite volume Runge-Kutta scheme (Jameson's algorithm) for Euler equations of gasdynamics. He has derived all the governing equations in a fully conservative nondimensionalized form suitable for discretization on a general nonorthogonal curvilinear computational grid.

Recently, Mr. Lee has successfully implemented the DMR concept in an explicit algorithm for the numerical integration of Navier-Stokes equations of laminar, incompressible flows through nozzles and cascades.

Another Ph.D. candidate, Mr. Daniel J. Dorney, worked on the analysis of existing numerical dissipation models and on physically based dissipation formulations for Euler and Navier-Stokes equations of gasdynamics. Mr. Lee and Mr. Dorney worked together on implementing DMR in both Euler and Navier-Stokes codes with the physically based artificial dissipation.

A Visiting Research Scientist, Mr. Ren Bing, was involved on the project at the Penn State University for six months. He performed a thorough survey of all Total Variation Diminishing (TVD) type schemes for controlling numerical dissipation.

III. LIST OF PUBLICATIONS AND PRESENTATIONS

1. Huang, C.-Y., and Dulikravich, G. S., "Fast Iterative Algorithms Based on Optimal Explicit Time-Stepping," Computer Methods in Applied Mechanics and Engineering, Vol. 63, August 1987, pp. 15-36.
2. Lee, S., Dulikravich, G. S., and Dorney, J. D., "Distributed Minimal Residual (DMR) Method for Explicit Algorithms Applied to Nonlinear Systems," presented at the Conference on Iterative Methods for Large Linear Systems, Austin, TX, Oct. 19-21, 1988.
3. Dulikravich, G. S., Dorney, J. D. and Lee, S., "Iterative Acceleration and Physically Based Dissipation for Euler Equations of Gasdynamics," presented at the ASME WAM '88, Symposia on Advances and Applications in Computational Fluid Dynamics, Chicago, IL, Nov. 28 - Dec. 2, 1988.
4. Lee, S., Dulikravich, G. S., and Dorney, J. D., "Acceleration of Iterative Algorithms for Euler Equations of Gasdynamics," AIAA paper 89-0097 presented at the AIAA Aerospace Sciences Meeting, Reno, NV, January 8-12, 1989.
5. Lee, S. and Dulikravich, G. S., "Acceleration of Iterative Algorithms for Euler Equations of Gasdynamics," (in press: AIAA Journal).
6. Lee, S., and Dulikravich, G. S., "Accelerated Computation of Viscous, Steady, Incompressible Flows," accepted for presentation at the ASME International Gas Turbine Conference, Toronto, Canada, June 4-8, 1989.
7. Lee, S., and Dulikravich, G. S., "A Fast Algorithm for High Reynolds Number Incompressible Navier-Stokes Equations," submitted for presentation at the 5th International Symposium on Numerical Methods in Engineering, Lausanne, Switzerland, Sept. 11-15, 1989.
8. Lee, S., and Dulikravich, "Acceleration of Implicit Algorithms Using Distributed Minimal Residual Method," submitted for presentation at the 2nd World Congress on Computational Mechanics, Stuttgart, F. R. Germany, August 1990.

IV. PROFESSIONAL PERSONNEL INVOLVED:

1. Dr. George S. Dulikravich, Associate Professor
2. Seungsoo Lee, Graduate Research Assistant, Ph.D. Candidate.
3. Daniel J. Dorney, Lecturer & Graduate Student, Ph.D. Candidate.
4. Ren Bing, Visiting Research Scientist, P.R.C. (05/87 - 11/87).

V. ADVANCED DEGREES AWARDED

1. Chung-Yuan Huang, "Optimization of Explicit Time-Stepping Algorithms and Stream-Function-Coordinate (SFC) Concept for Fluid Dynamics Problems," Ph.D. Dissertation, University of Texas at Austin, May 1987.
2. Stephen R. Kennon, "Numerical Solution of Weak Forms of Conservation Laws on Optimal Unstructured Triangular Grids," Ph.D. Dissertation, University of Texas at Austin, August 1987.

VI. INTERACTIONS (COUPLING ACTIVITIES)

1. Lecture delivered at the Computational Fluid Dynamics Branch/Institute for Computational Mechanics in Propulsion of the NASA Lewis Research Center in Cleveland, Ohio in May, 1987. Title: "New Concepts in Computational Fluid Dynamics." Speaker: Prof. G. S. Dulikravich.
2. Dulikravich, G. S., "Optimization of Explicit Multi-Step Algorithms," paper presented at the First International Conference on Industrial and Applied Mathematics, Paris, France, June 29-July 3, 1987.

3. A two-day workshop at the University of Rijeka, Rijeka, Yugoslavia in July, 1987. Title: "Methods of Computational Fluid Dynamics for Turbomachinery," speakers: Prof. G. S. Dulikravich and Prof. L. J. Hayes.
4. A one-day visit to Ecole Polytechnique outside of Paris, France in July, 1987 and a discussion concerning transonic potential flow computations and influence of different artificial dissipation models. Prof. G. S. Dulikravich.
5. A one-day visit to Ecole Polytechnique Federale de Lausanne in Switzerland in July, 1987. Discussion of mutually interesting research on prediction of water flow fields in hydroturbines and the effects of artificial dissipation on the results of hydrocodes. Prof. G. S. Dulikravich.
6. Invited lecture delivered at the Department of Aerospace Engineering, University of Colorado, Boulder, CO in January 1988. Title: "Numerical Dissipation, Grid Generation and Fast Iterative Algorithms." Speaker: Prof. G. S. Dulikravich.
7. Invited one-day workshop (G. S. Dulikravich was the only lecturer) on "Inverse Design and Special Topics in Computational Fluid Dynamics," United Technologies Research Center, Hartford, CT, March 1988.
8. Lecture, Inst. for Computational Methods in Propulsion, NASA Lewis Research Center, Cleveland, OH, May 1988.
9. Invited Lecture, Turboinstitut, Ljubljana, Yugoslavia, Sept. 1988.
10. Invited Lecture, Department of Mechanical Engineering, University of Texas, Austin, TX, Oct. 1988.
11. Invited Lecture, Department of Aerospace Engineering, Ohio State University, Columbus, OH, Oct. 1988.

VII. NEW DISCOVERIES, INVENTIONS AND PATENTS

Although original, this type of work is not patentable. Consequently, there were no discoveries, inventions or patents resulting from this research project.

VIII. SUGGESTIONS FOR FUTURE RESEARCH

We also developed a DMR version of a transonic Navier-Stokes finite volume code for two-dimensional shock/boundary layer interaction. In addition, Mr. Lee implemented DMR in a two-dimensional implicit ADI (Beam-Warming) solver for incompressible Navier-Stokes equations. He also developed a fully three dimensional DMR version of an explicit code for incompressible Navier-Stokes equations. These three codes remain to be tested especially when using DMR formulation with implicit algorithms.

Notice that all numerical results with the DMR method were obtained without the standard acceleration techniques such as explicit and implicit residual smoothing, enthalpy damping, multigridding and vectorization. These methods could be combined with the DMR to further accelerate the iterative algorithms.

Furthermore, it would be highly desirable to study the effect of grid clustering, grid size, grid orthogonality and grid structure on the DMR. In addition, domain partitioning could be used with different DMR sequences in each domain thus leading to accelerated parallel processing capabilities.

COMPUTER METHODS IN APPLIED MECHANICS AND ENGINEERING 63 (1987) 15-36
NORTH-HOLLAND

FAST ITERATIVE ALGORITHMS BASED ON OPTIMIZED EXPLICIT TIME STEPPING

Chung-Yuan HUANG
Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712, U.S.A.

George S. DULIKRAVICH
Department of Aerospace Engineering, Penn State University, University Park, PA 16802, U.S.A.

Received 9 May 1986
Revised manuscript received 26 September 1986

The Generalized Nonlinear Minimal Residual (GNLMR) method is shown to consistently accelerate and stabilize iterative algorithms for solving nonlinear problems by using the optimized explicit multistepping. The examples presented in this paper illustrate the beneficial effects of the optimized multistep algorithm on the computational efficiency and the convergence rate as applied to several nonlinear problems in fluid dynamics. The significant reduction in computing time when using the multiple optimized acceleration factors is only negligibly weighed down by the computation costs due to the requirements for additional computer storage.

1. Introduction

The relaxation factor used in accelerating an iterative method to obtain the converged solution plays the same role as the time step size in advancing the transient solution to the steady-state solution for a time-dependent problem. The classical analyses for the stability of numerical schemes for solving time-dependent problems neglect boundary conditions and assume a uniform computational grid. Furthermore, these analyses are based on linear equations with constant coefficients and the assumptions of small perturbations and applicability of Fourier analysis [1, 2]. However, Cheng [2] pointed out that the perturbation of the error in the finite difference calculations may not be small and that the error in the finite difference calculations may not satisfy the conditions for Fourier series expansion. In addition, Mitchell and Griffiths [1] pointed out that the errors due to approximate or additional boundary conditions are represented by modes which are not of Fourier type. Thus, the linear stability analysis usually results in overly restrictive and even incorrect conclusions.

The numerical experiments performed by Kennon and Dulikravich [3] and Kennon [4] using the NonLinear Minimal Residual (NLMR) method showed that the usual Courant–Friedrichs–Lewy (CFL) number limitation for both linear and nonlinear problems can be significantly exceeded. The NLMR method provided a simple analytic way to determine the

5

optimal acceleration factors for both linear and nonlinear problems. However, the elementary time steps used for obtaining the corrections still follow the CFL number limitation concluded from the linear analysis.

The generalized nonlinear minimal residual (GNLMR) method developed by Huang, Kennon and Dulikravich [5] provided a practical analytical tool to determine the exact stability conditions for both linear and nonlinear problems in arbitrary domains. If accurate time evolution is required when solving an unsteady problem, the limitation on the time step size can be analytically determined by using the GNLMR method. If transient behavior is of no interest, the GNLMR method can be applied to determine the optimal value of the time step size (optimal acceleration factor) to minimize the number of time steps (number of iterations) for obtaining the steady-state converged solution.

The main objective of this paper is to investigate the effects of the optimized multistep algorithm on the computational efficiency and on the monotonicity of convergence rate of the GNLMR method. The analytic investigation is confirmed on four nonlinear test cases: the one-dimensional and two-dimensional viscous Burgers' equations and the two-dimensional incompressible and compressible Stream-Function-Coordinate (SFC) equations [6].

2. Theoretical aspects

2.1. Multistep minimal residual method for linear problems

Let us first consider a well-posed linear initial value problem:

$$\begin{aligned} &\partial\varphi/\partial\tau = L\varphi - F && \text{in } \Omega\,, \\ &\varphi = \varphi_B && \text{on } \partial\Omega\,, \\ &\varphi = \varphi_0 && \text{at } \tau = \tau_0\,. \end{aligned} \tag{1}$$

Define

$$r^t = l\varphi^t - f \tag{2}$$

as the residual vector at time level t. Here, l denotes the scheme-dependent discrete analog of L, f is the discrete analog of F and also includes boundary terms.

Assume that M steps are used to iterate at each time level t. Using the Einstein summation convention where repeated subscripts are summed, the multistep algorithm for (1) is then defined as follows:

$$\varphi^{t+1} = \varphi^t + \omega_m \delta_m\,, \quad m = 1, 2, \ldots, M\,, \tag{3}$$

where

$$\begin{aligned} &\delta_1 = l\varphi^t - f\,, \\ &\delta_m = l^{m-1}(\delta_1)\,, \quad m > 1\,, \end{aligned} \tag{4}$$

are the corrections at step m. Coefficients ω_m are the corresponding relaxation factors to be determined by minimizing the L_2 norm of the residual at time level $(t+1)$. With the definition

of residual vector (2) and the help of (3), the following relation can easily be verified:

$$r^{t+1} = r^t + \omega_m l\delta_m = r^t + \omega_m l^m r^t . \tag{5}$$

The L_2 norm of the residual vector at time level $(t+1)$ is then

$$\|r^{t+1}\|^2 = \|r^t\|^2 + 2\omega_m (r^t, l\delta_m) + (l\delta_m, l\delta_n)\omega_m \omega_n , \quad m, n = 1, 2, \ldots, M . \tag{6}$$

It should be pointed out that the boundary condition for the corrections δ_m in (3) is apparently zero. However, the boundary conditions for the residual vector r^t and corrections δ_m in (5) can be determined either by extrapolation from the interior points or simply by setting them equal to zero. The residual norm will then converge to the norm of the truncation error of the difference scheme if the first method is applied and to the machine accuracy if the second method is applied.

The highest rate of convergence is possible when ω_m are the solutions of the following system of linear equations:

$$\partial \Gamma / \partial \omega_m = 0 \quad \text{or} \quad (r^t, l\delta_m) + (l\delta_m, l\delta_n)\omega_n = 0 , \tag{7}$$

where the rate of convergence Γ is defined as

$$\Gamma = -\log(\|r^{t+1}\| / \|r^t\|) . \tag{8}$$

Multiplying (7) by ω_m, it follows that

$$(r^t, l\delta_m)\omega_m + (l\delta_m, l\delta_n)\omega_m \omega_n = 0 . \tag{9}$$

Subtracting (9) from (6) and using (7) results in

$$\|r^{t+1}\|^2 - \|r^t\|^2 = (r^t, l\delta_m)\omega_m = -(l\delta_m, l\delta_n)\omega_m \omega_n = -\int_\Omega (\omega_m l\delta_m)^2 \, \mathrm{d}\Omega < 0 . \tag{10}$$

Thus, the residual norms for the multistep minimum residual method show a monotone convergence behavior which guarantees the stability of the iterative scheme and produces the highest rate of its convergence.

2.2. Optimization of the Euler scheme for nonlinear problems

For clarity, we consider two-dimensional problems and equations in conservative form only. The extension to multidimensional problems and nonconservative equations is then straightforward.

The conservative form of the governing equations for most engineering problems can be written as:

$$\partial \varphi / \partial \tau = L_\nu N^\nu(\varphi, \varphi_x, \varphi_y) - F , \tag{11}$$

7

where the operators are

$$L_1 = \partial/\partial x\,, \qquad L_2 = \partial/\partial y\,,$$

and N^{ν} is the nonlinear differential operator in coordinates x_i. Using the Euler one-step, time-consistent, explicit scheme, the finite difference form of (11) can be written as:

$$\varphi^{t+1} = \varphi^t + \Delta\tau r^t\,, \tag{12}$$

where

$$r^t = l_\nu N^\nu(\varphi^t, \varphi^t_x, \varphi^t_y) - f \tag{13}$$

is defined as the residual at time level t. Therefore, the residual at time level $(t+1)$ can be expressed as

$$r^{t+1} = l_\nu N^\nu(\varphi^{t+1}, \varphi^{t+1}_x, \varphi^{t+1}_y) - f\,. \tag{14}$$

After expanding the nonlinear discretized operator N^ν in a Taylor series, it follows that

$$r^{t+1} = l_\nu\{N^\nu(\varphi^t, \varphi^t_x, \varphi^t_y) + [(\partial N^\nu/\partial\varphi^t))r^t + (\partial N^\nu/\partial\varphi^t_x)(r^t)_x$$
$$+ (\partial N^\nu/\partial\varphi^t_y)(r^t)_y]\Delta\tau + \mathrm{O}(\Delta\tau^2)\} - f\,. \tag{15}$$

In summary,

$$r^{t+1} = r^t + a_p(\Delta\tau)^p\,, \quad 1 \leqslant p \leqslant P\,, \tag{16}$$

where P is the degree of the nonlinearity of the operator N. Equation (16) indicates that the residual at time level $(t+1)$ is a polynomial (henceforth called Residual Polynomial [3] or RP) of the time step size, $\Delta\tau$. Thus, the L_2 norm of the residual at time level $(t+1)$ can be expressed as

$$\|r^{t+1}\|^2 = \|r^t\|^2 + 2(r^t, a_p)(\Delta\tau)^p + (a_p, a_q)(\Delta\tau)^p(\Delta\tau)^q\,, \quad 1 \leqslant p\,,\ q \leqslant P\,. \tag{17}$$

Equation (17) implies that the residual norm at time level $(t+1)$ is a positive polynomial (henceforth called Minimizing Polynomial [3] or MP) of the time step size $\Delta\tau$, which is to be determined. Thus, the convergence of scheme (12) will be guaranteed provided that $\Delta\tau$ is chosen in such a way that $\Gamma > 0$. The highest rate of convergence can be achieved only when $\Delta\tau$ is chosen as the optimizer of the minimizing polynomial (17) such that $\|r^{t+1}\|$ is an infimum. However, the determination of the optimizer needs special numerical techniques [7]. To avoid this difficulty, the linearized operator [3–5] of N^ν may be applied. If N^ν is truncated to the first order in $\Delta\tau$ (linearized operator), the approximate residual vector is

$$r^{*t+1} = r^t + a_1\Delta\tau\,, \tag{18}$$

where

$$a_1 = l_\nu[(\partial N^\nu/\partial\varphi^t)r^t + (\partial N^\nu/\partial\varphi^t_x)(r^t)_x + (\partial N^\nu/\partial\varphi^t_y)(r^t)_y]\,. \tag{19}$$

8

Then, the approximate MP is

$$\|r^{*t+1}\|^2 = \|r^t\|^2 + 2(a_1, r^t)\Delta\tau + (a_1, a_1)(\Delta\tau)^2 . \tag{20}$$

The optimal time step for the explicit Euler scheme can be easily found as

$$(\Delta\tau)_{\text{opt.}} = -(a_1, r^t)/\|a_1\|^2 . \tag{21}$$

2.3. The generalized nonlinear minimum residual (GNLMR) method

The GNLMR method actually is the application of the methods described in the previous sections. The multistep algorithm for nonlinear problems is defined as

$$\varphi^{t+1} = \varphi^t + \omega_m \delta_m + O(\omega_m^2) , \quad m = 1, 2, \ldots, M , \tag{22}$$

where repeated indices are summed. The correction at the first step if defined as

$$\delta_1 = r^t = l_\nu N^\nu(\varphi^t, \varphi^t_x, \varphi^t_y) - f . \tag{23}$$

The correction at step $m > 1$ is defined as

$$\delta_m = l_\nu[(\partial N^\nu/\partial\varphi^t)\delta_{m-1} + (\partial N^\nu/\partial\varphi^t_x)(\delta_{m-1})_x + (\partial N^\nu/\partial\varphi^t_y)(\delta_{m-1})_y] . \tag{24}$$

The coefficients of the higher-order terms of ω_m can be obtained by Taylor-series expansion. If only linear terms of ω_m are retained, the residual polynomial (RP) at time level $(t + 1)$ can be expressed by Taylor-series expansion as

$$\begin{aligned} r^{t+1} &= l_\nu N^\nu(\varphi^{t+1}, \varphi^{t+1}_x, \varphi^{t+1}_y) - f \\ &= l_\nu N^\nu[\varphi^t + \omega_m\delta_m, \varphi^t_x + \omega_m(\delta_m)_x, \varphi^t_y + \omega_m(\delta_m)_y] - f \\ &= r^t + l_\nu\{[(\partial N^\nu/\partial\varphi^t)\delta_m + (\partial N^\nu/\partial\varphi^t_x)(\delta_m)_x + (\partial N^\nu/\partial\varphi^t_y)(\delta_m)_y]\omega_m + O(\omega_m^2)\} . \end{aligned} \tag{25}$$

Therefore, the minimizing polynomial (MP) at time level $(t + 1)$ can be determined as

$$\|r^{t+1}\|^2 = \|r^t\|^2 + g(\omega_m) , \tag{26}$$

where $g(\omega_m)$ is a polynomial in ω_m. For a highly nonlinear differential equation, g will be a complicated multivariable polynomial that depends on the total number of intermediate steps M that were used and the degree of the nonlinearity of the differential operator N^ν. Thus, a fast and accurate procedure of determining the optimizer of MP is required for the GNLMR method to guarantee the highest rate of convergence. If the linearized operator of N^ν is used, the method that was described in Section 2.1 can be applied to determine the approximate optimizer of (26).

The GNLMR method requires $(M + 1)$ times larger computer storage to save the correc-

9

tions from M intermediate steps than does the single-step nonaccelerated scheme. Some additional algebraic operations are also required to determine the coefficients of the MP which are obtained by integrating the corrections over the entire domain. The storage requirement of the GNLMR method is quite acceptable when compared with the excessive storage required by the GMRES method [8, 9]. It should be pointed out that the GMRES method also needs a large number of arithmetic operations not only for orthonormalizing search directions but also for determining the optimal weighing parameters in updating the iterative solutions.

3. Numerical examples

Four test cases were used to demonstrate the application, the computational efficiency, and the monotone convergence behavior of the GNLMR method. Since it was found [5] that the linearized residual polynomial still guarantees a relatively high convergence rate, it will be used for all test cases. The first two cases representing the one-dimensional and two-dimensional viscous Burgers' equation were solved by the time-dependent technique as described in Section 2.

The last two test cases, the two-dimensional incompressible and compressible stream-function-coordinate (SFC) equations [6] were solved in their steady-state and nonconservative form. Liebman's or Gauss–Seidel's method was applied to determine the correction at each intermediate iterative step m.

Details about the control parameters such as grid size, stopping criteria, and number of acceleration factors used for each test case are summarized in Table 1. For all test cases, comparisons are based on the relative improvement of computational efficiency that can be

Table 1
Summary of the control parameters for numerical test cases

Test case	Max. no. of ω used	Boundary conditions for residual and corrections	Stopping criteria	Grid size
Case 1: 1D Burgers' equation	8	Extrapolation from interior data Zero	$\Gamma \leq 10^{-8}$ $\lVert r' \rVert \leq 10^{-8}$	41
Case 2: 2D Burgers' equation	8	Extrapolation from interior data Zero	$\Gamma \leq 10^{-8}$ $\lVert r' \rVert \leq 10^{-8}$	51 × 51
Case 3: 2D incomp. SFC equation	8	Zero	$\lVert r' \rVert \leq 10^{-8}$	47 × 11
Case 4: 2D comp. SFC equation	8	Zero	$\lVert r' \rVert \leq 10^{-8}$	61 × 11

10

obtained using different total number of intermediate steps M. The relative improvement of computational efficiency η_M is defined as

$$\eta_M = T_0 / T_M \,, \tag{27}$$

where T_0 denotes the computing time spent for the nonaccelerated method and T_M denotes the computing time spent for the acceleration method based on the stopping criteria as described in Table 1. The results are summarized in Fig. 9.

3.1. Burgers' equations

According to the notations defined in Section 2, the one-dimensional, viscous Burgers' equation can be written as

$$\partial\varphi / \partial\tau = \partial / \partial x[N(\varphi, \varphi_x)] \,, \tag{28}$$

where

$$N(\varphi, \varphi_x) = -\tfrac{1}{2}\varphi^2 + \nu\varphi_x \,, \tag{29}$$

and ν is the viscosity coefficient. In this example $\nu = 0.07$ is used. The initial and the boundary conditions are chosen as follows:

$$\varphi(1, \tau) = 0 \,, \qquad \varphi(0, \tau) = 1 \,, \qquad \varphi(x, 0) = 1 - x \,. \tag{30}$$

The two-dimensional, nonlinear, viscous Burgers' equation solved by Ghia et al. [10] was chosen in the presented test case. According to the notations defined in Section 2, it can be expressed as

$$\partial u / \partial\tau = \partial / \partial x[N^1(u, u_x)] + \partial / \partial y[N^2(u, u_y)] \,, \tag{31}$$

where

$$N^1(u, u_x) = u_x - \lambda y(\tfrac{1}{2}u^2 - uU) \,, \qquad N^2(u, u_y) = u_y - \lambda x(\tfrac{1}{2}u^2 - uU) \,. \tag{32}$$

Here, λ is a parameter and U is a constant. The values of U and λ used in this test case are 0.5 and 2.0, respectively.

The FTCS scheme is applied to discretize (28) and (31). If the linearized form of operator N is used with M steps at each time level t, the residual polynomial is truncated up to its first order as

$$\mathrm{RP} = r^{t+1} = r^t + a_m \omega_m \,, \tag{33}$$

where

$$a_m = \partial / \partial x[-\varphi^t \delta_m + \nu(\delta_m)_x] \tag{34}$$

for the one-dimensional case and

$$a_m = \partial / \partial x[(\partial N^1 / \partial u)\delta_m + (\partial N^1 / \partial u_x)(\delta_m)_x] + \partial / \partial y[(\partial N^2 / \partial u)\delta_m + (\partial N^2 / \partial u_y)(\delta_m)_y] \tag{35}$$

11

for the two-dimensional case. The corrections δ_m at each intermediate step m can be determined by (23), (24). The minimizing polynomial MP for both cases is then

$$\mathrm{MP} = \|r^{t+1}\|^2 = \|r^t\|^2 + 2(r^t, a_m)\omega_m + (a_m, a_n)\omega_m\omega_n \,. \tag{36}$$

Thus, the optimal acceleration parameters ω_m can be easily determined by solving the following system of linear equations:

$$A_{mn}\omega_n = b_m \,, \tag{37}$$

where

$$A_{mn} = (a_m, a_n) \,, \tag{38}$$

$$b_m = -(r^t, a_m) \,, \tag{39}$$

and A_{mn} is a symmetric matrix of order M.

The boundary conditions of the residual and corrections in (33) were determined by extrapolating them from the interior points (Case a) or by explicitly enforcing them to be zero (Case b). The stopping criteria for all cases were such that the computations were terminated when the asymptotic rate of convergence (Case a) or the norm of the residual (Case b) approached the machine accuracy.

It must be mentioned that the norm of the truncation error represents the maximum attainable accuracy of a numerical scheme and is obviously scheme-dependent. It can be seen from Figs. 2(a) and 4(a) that under the same stopping criteria the residual norms for all cases converge to the values corresponding to the respective norms of the truncation error. Moreover, Figs. 1(a) and 2(a) illustrate that the accuracy of the nonaccelerated scheme can be improved by applying the GNLMR method.

Since the linearized operators were used in these two test cases, the convergence history shown in Figs. 1–4 exhibit a similar behavior as in the linear problems as solved in our earlier works [5]. It is obvious that if the GNLMR method is applied, the number of iterations and the computing time required to achieve the asymptotic rate of convergence are considerably lowered as compared to the nonaccelerated schemes (Figs. 1(a), 2(a), 3(a) and 4(a)). Moreover, the time required by the GNLMR method for marching the solution from the asymptotic state to a fully converged solution is much shorter than with the nonaccelerated method. The improvement of the computational efficiency that can be obtained using a different number of intermediate steps M is summarized in Fig. 9.

Although the computational efficiency increases significantly with the increasing number of intermediate steps M, the improvement becomes less pronounced and even shows a reverse trend after approximately $M = 5$ in the one-dimensional case. The reason for this unexpected result is that when using a multistep algorithm, an $M \times M$ matrix has to be inverted (directly) at each time level, t. The number of operations and computing time required for the direct inversion of a matrix grows very fast with the increase of the matrix size, thus countering the benefits of adding more intermediate steps in the multistep procedure especially for one-dimensional problems.

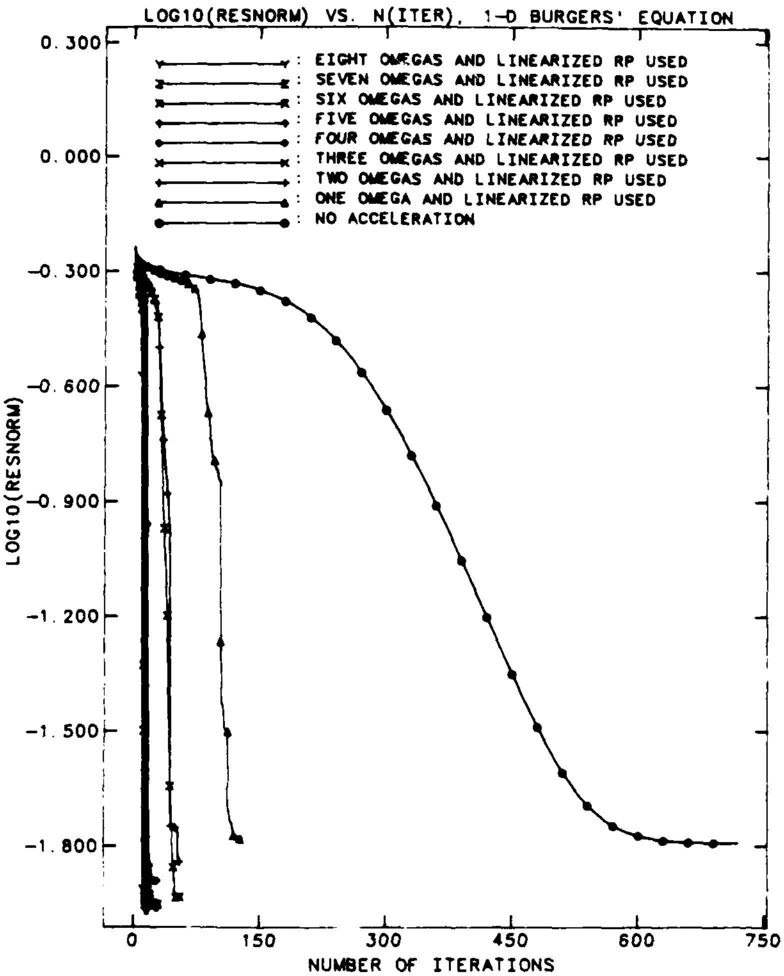


Fig. 1(a). Residual norm versus the number of iterations, 1D Burgers' equation.

3.2. *Stream-function-coordinate (SFC) equations*

The two-dimensional stream-function-coordinate (SFC) equation for an irrotational, inviscid, steady flow derived by Huang and Dulikravich [6] is given by:

$$(y_\psi^2 - \sigma)y_{xx} - 2y_x y_\psi y_{x\psi} + (1 + y_x^2)y_{\psi\psi} = 0\,, \tag{40}$$

where σ represents the compressibility and is equal to zero for incompressible flows. It is defined as

$$\sigma = (\rho^* a^* / \rho a)^2 = [\tfrac{1}{2}(\gamma + 1) - \tfrac{1}{2}(\gamma - 1)M^{*2}]^{-(\gamma+1)/(\gamma-1)}\,. \tag{41}$$

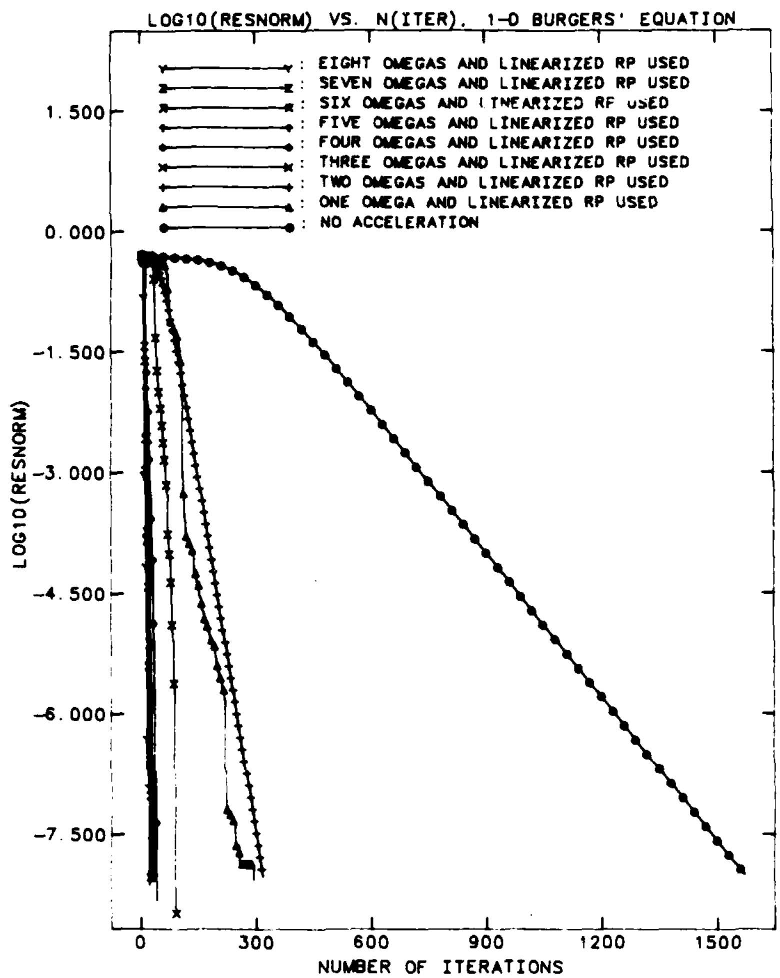


Fig. 1(b). Residual norm versus the number of iterations, 1D Burgers' equation.

where ρ and a denote the local density and the local speed of sound, respectively, and γ is the ratio of the specific heats. The superscript terms denote the characteristic quantities of the flow. It can be shown [6] that σ is an implicit function of y_x and y_ψ, that is

$$(1 + y_x^2)/y_\psi^2 = [(\gamma + 1)\sigma^{(\gamma-1)/(\gamma+1)} - 2]/[(\gamma - 1)\sigma] . \tag{42}$$

Let us define

$$\begin{aligned} &c_1 = y_x , \quad c_2 = y_\psi , \quad c_3 = y_{xx} , \\ &c_4 = y_{x\psi} , \quad c_5 = y_{\psi\psi} . \end{aligned} \tag{43}$$

Then (40) can be rewritten as

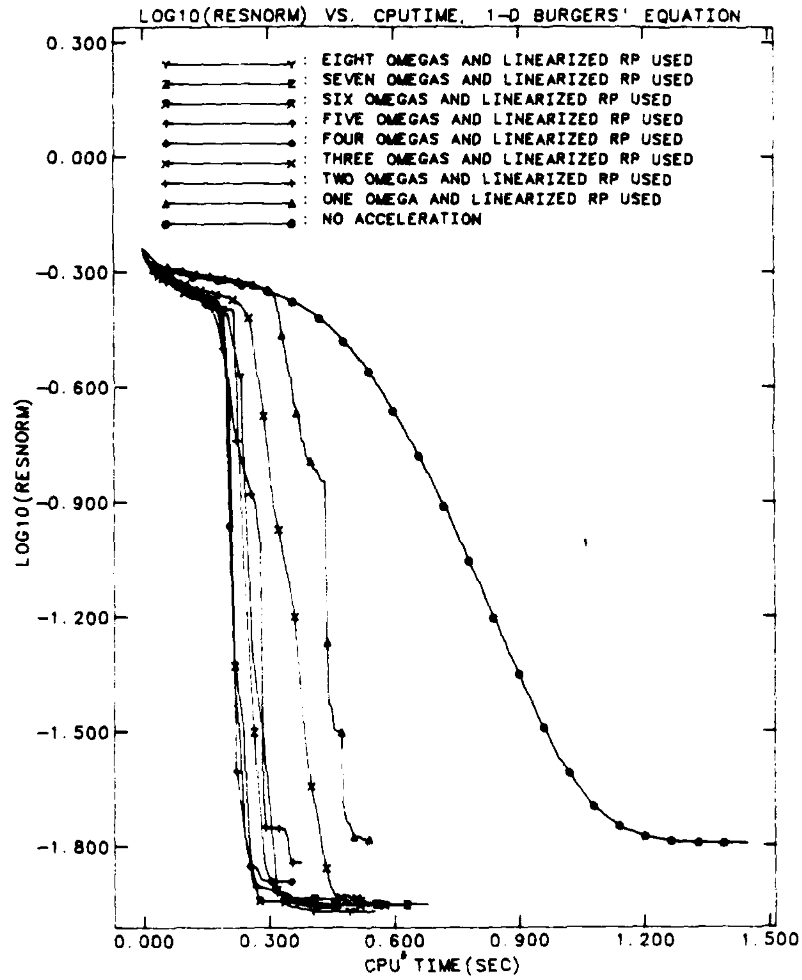


Fig. 2(a). Residual norm versus the computing time, 1D Burgers' equation.

$$N(c_1, c_2, c_3, c_4, c_5) = 0 , \tag{44}$$

or

$$[c_2^2 - \sigma(c_1, c_2)]c_3 - 2c_1c_2c_4 + (1 + c_1^2)c_5 = 0 . \tag{45}$$

Assume that a uniform computational grid is used (both Δx and $\Delta\psi$ are constant) and central differencing is applied to discretize all derivatives. The finite difference approximation of the SFC equation can be expressed as

$$y_{i,j} = [(y_\psi^2 - \sigma)/\Delta x^2 + (1 + y_x^2)/\Delta\psi^2]^{-1}[(y_\psi^2 - \sigma)(y_{i+1,j} + y_{i-1,j})/(2\Delta x^2) + (1 + y_x^2)(y_{i,j+1} + y_{i,j-1})/(2\Delta\psi^2) - y_x y_\psi y_{x\psi}] . \tag{46}$$

15

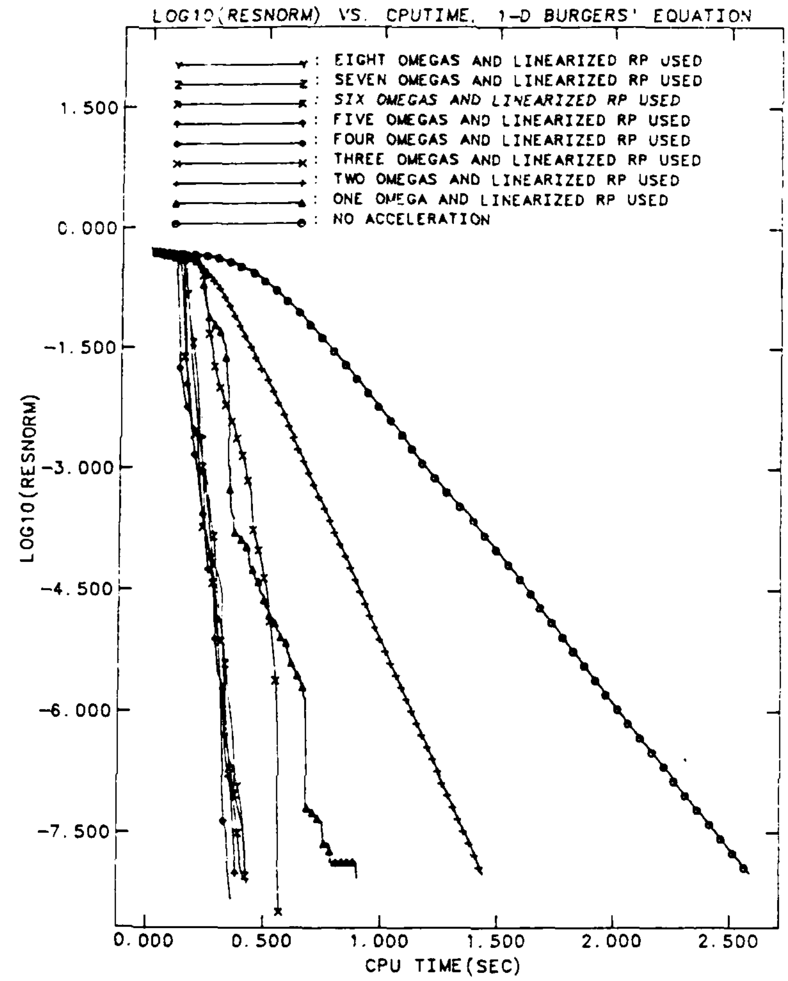


Fig. 2(b). Residual norm versus the computing time, 1D Burgers' equation.

Equation (46) will be referred to as the iterative equation. Most iterative schemes for solving (46) can be expressed as

$$y_{i,j}^{t+1} = y_{i,j}^{t} + \omega\delta_{i,j}^{t} , \tag{47}$$

where t represents the iteration level, ω is the relaxation factor, and $\delta_{i,j}^{t}$ is the correction at iteration level t. It is defined as

$$\delta_{i,j}^{t} = \bar{y}_{i,j}^{t+1} - y_{i,j}^{t} . \tag{48}$$

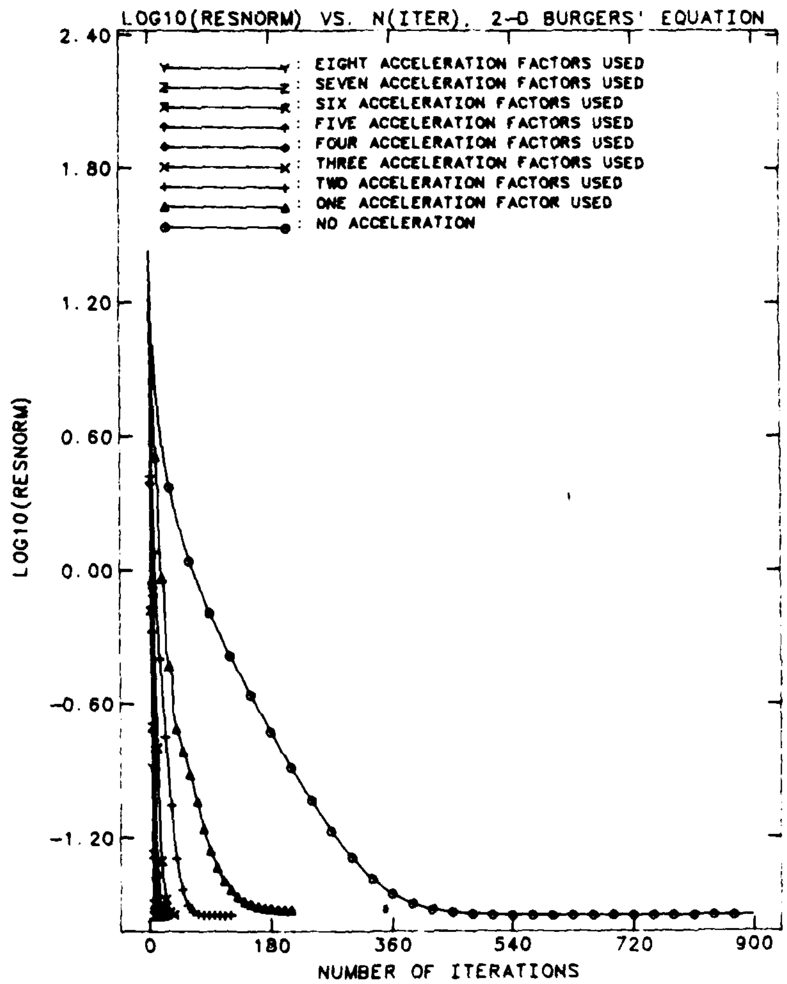


Fig. 3(a). Residual norm versus the number of iterations. 2D Burgers' equation.

Here, $\bar{y}_{i,j}^{t+1}$ is the temporary value of y at iteration level $(t+1)$ obtained by applying (46) with any fundamental iterative scheme. In the presented studies, Liebman's method ($\omega = 1$) was used as the fundamental iterative scheme and will be henceforth referred to as the nonaccelerated method. Assume that M steps are used in the GNLMR method. The solution is then updated by using

$$y_{i,j}^{t+1} = y_{i,j}^{t} + \omega_m \delta_m , \quad m = 1, 2, \ldots, M , \tag{49}$$

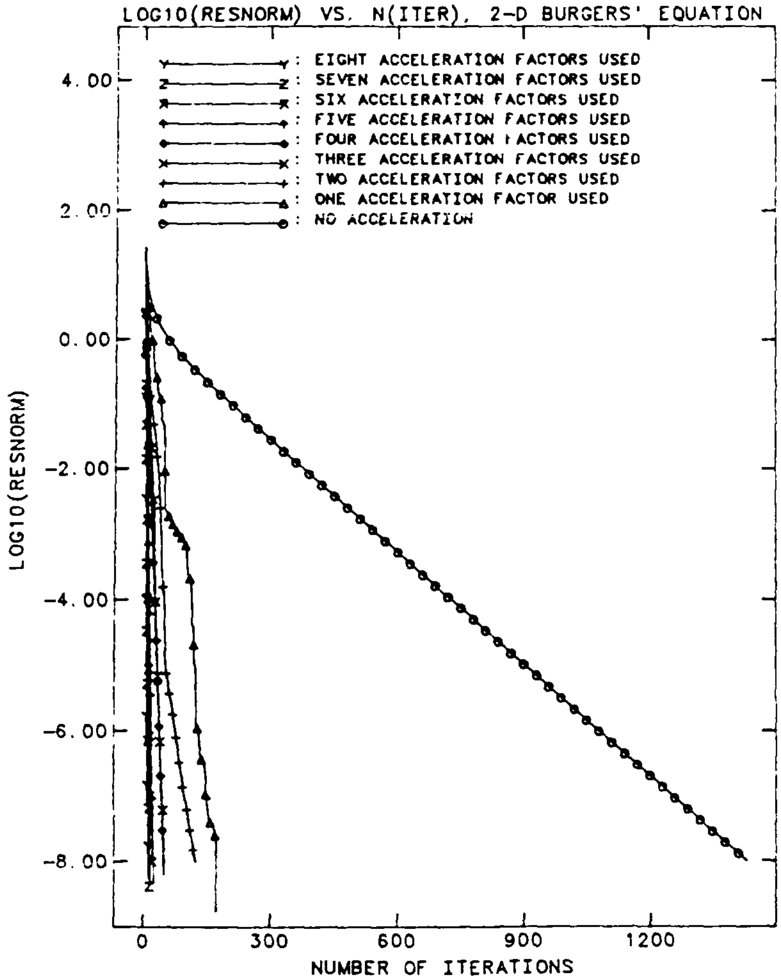


Fig. 3(b). Residual norm versus the number of iterations, 2D Burgers' equation.

where δ_m are the corrections at each intermediate step m. They are obtained by successively applying Liebman's method. The optimal values of ω_m based on a linearized RP can be determined by solving (37) with

$$r^t = N(c_1, c_2, c_3, c_4, c_5)^t , \tag{50}$$

$$\begin{aligned} a_m = [&(\partial N/\partial c_1)(\delta_m)_x + (\partial N/\partial c_2)(\delta_m)_\psi + (\partial N/\partial c_3)(\delta_m)_{x\psi} \\ &+ (\partial N/\partial c_4)(\delta_m)_{xx} + (\partial N/\partial c_5)(\delta_m)_{\psi\psi}] . \end{aligned} \tag{51}$$

18

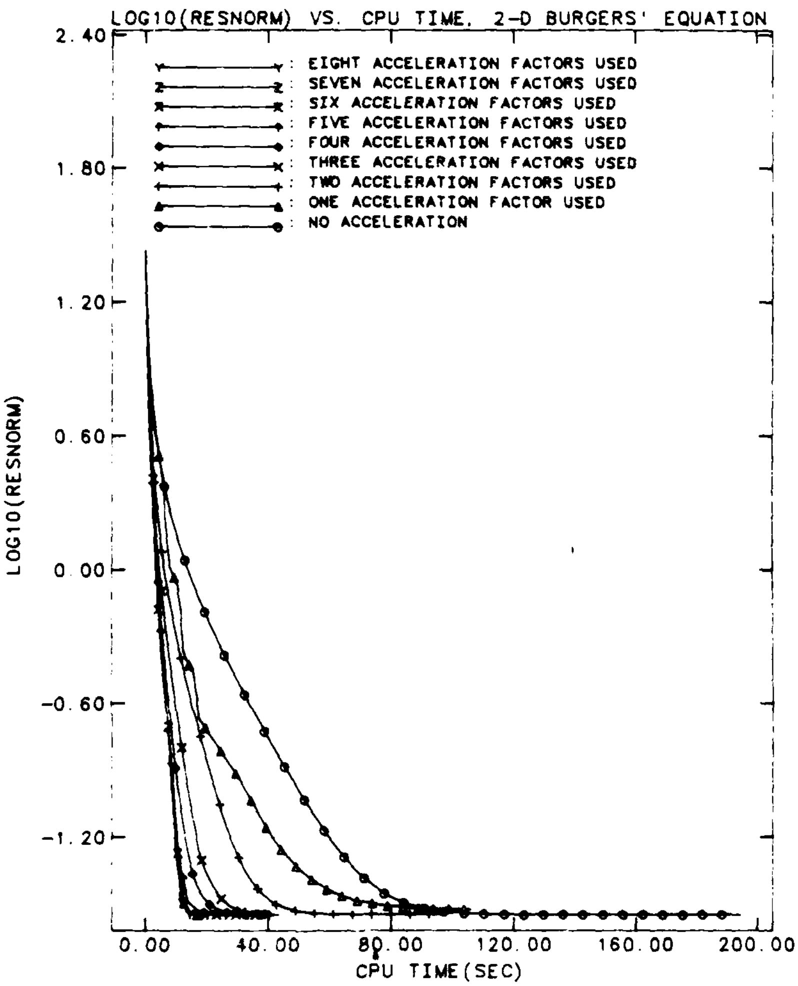


Fig. 4(a). Residual norm versus the computing time, 2D Burgers' equation.

For the incompressible case, a uniform flow around a cascade of doublets was solved [6]. For the compressible case, a subsonic flow with free-stream Mach number $M_\infty = 0.65$ around a NACA 0012 airfoil in a channel with height/chord ratio =3.6 was solved [6].

Since linearized operators were used in these two cases, and the boundary conditions for the residual and corrections in (33) were set to zero, the residual norm will converge to machine accuracy. Therefore, the stopping criteria for these two cases was chosen in such a

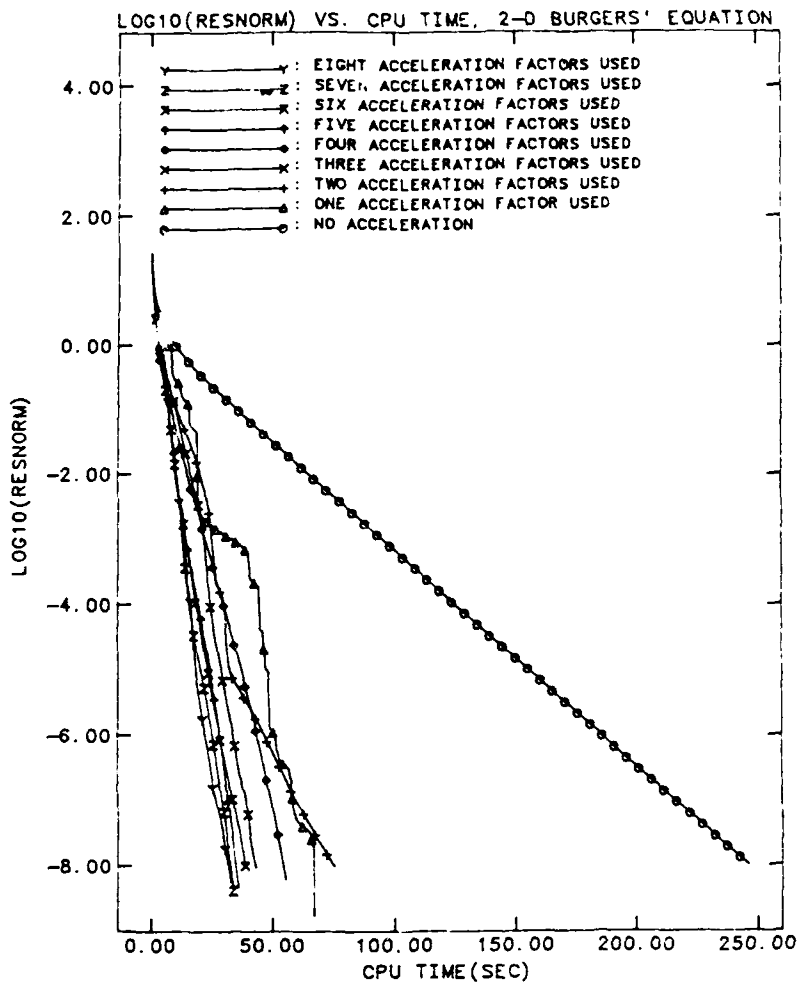


Fig. 4(b). Residual norm versus the computing time, 2D Burgers' equation.

way that as the residual norm approaches machine accuracy, the computation is forced to stop.

The improvement of the computational efficiency is summarized in Fig. 9. Both cases show that the computational efficiency is increased significantly by increasing the number of intermediate steps M.

The numerical results for these two cases are summarized in Figs. 5–8. Both cases exhibit a

20

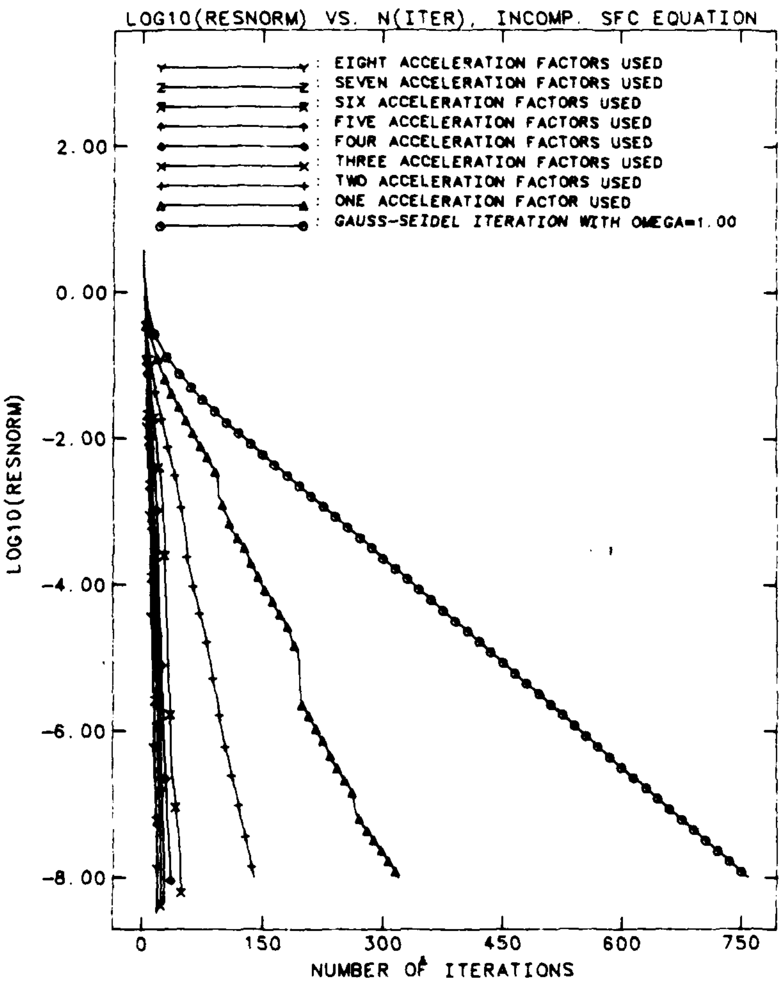


Fig. 5. Residual norm versus the number of iterations, incompressible SFC equation.

common feature that the monotonicity and the rate of convergence increase with the increase of the total number of intermediate steps M. Most importantly, Figs. 6 and 8 show that with a specified minimum computing time, the difference in the residual norms between the nonaccelerated method and the GNLMR method varies between one to eight orders of magnitude depending on the number of steps used in the GNLMR method. This fact strongly proves the computational efficiency that can be obtained using the GNLMR method [8].

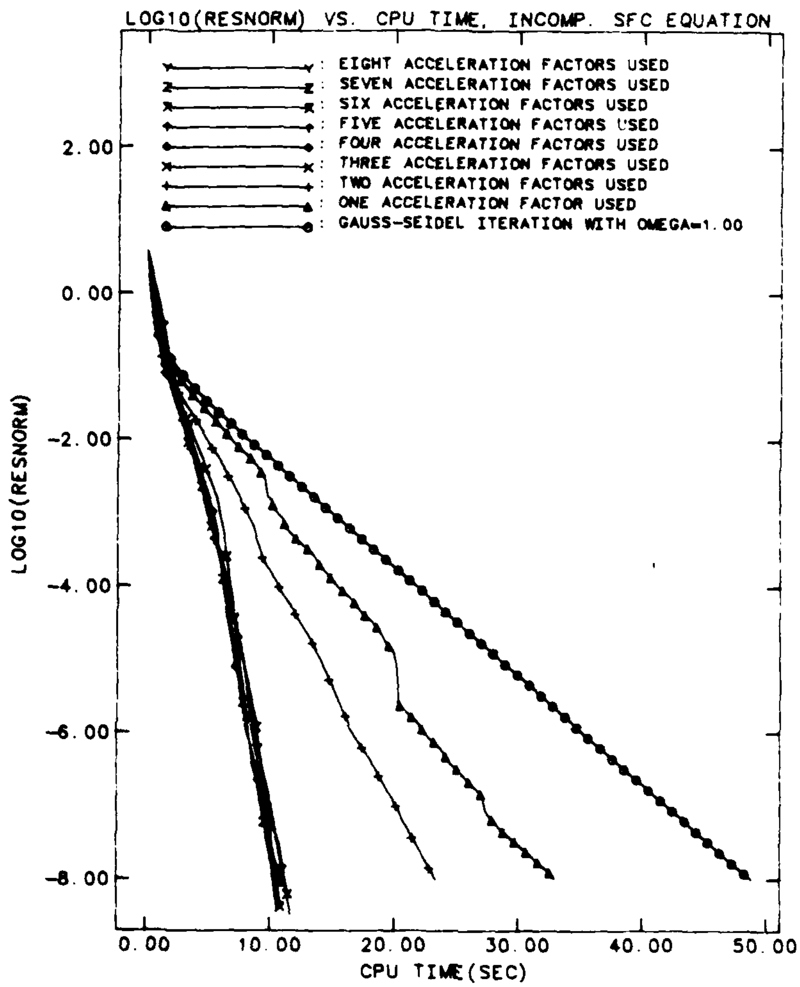


Fig. 6. Residual norm versus the computing time, incompressible SFC equation.

4. Concluding remarks

Four numerical test cases for nonlinear problems in fluid dynamics were presented to demonstrate the applicability, computational efficiency, and monotone convergence behavior of the GNLMR method. It was found that even though the theory of the GNLMR method is based on the evolution problems and equations in conservative form, the method can be applied equally successfully to the solutions of steady-state problems governed by equations in

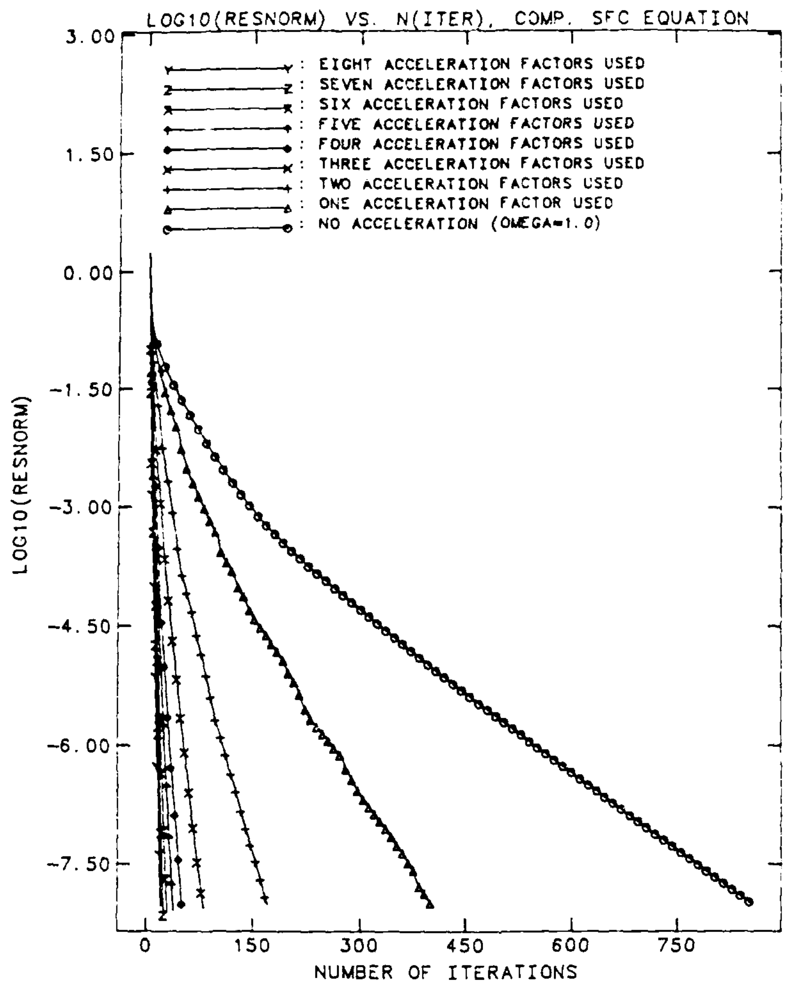


Fig. 7. Residual norm versus the number of iterations, compressible SFC equation.

nonconservative form. The results for all test cases show that when applying the GNLMR method to nonlinear problems, the number of iterations and the corresponding computer time are considerably lowered by increasing the number of intermediate time steps.

Since the explicit multistep algorithm was employed in developing the GNLMR method, the advantage of accelerating the convergence rate of the iterative process is partially offset by some extra costs. These are caused by the requirements for additional storage in order to save

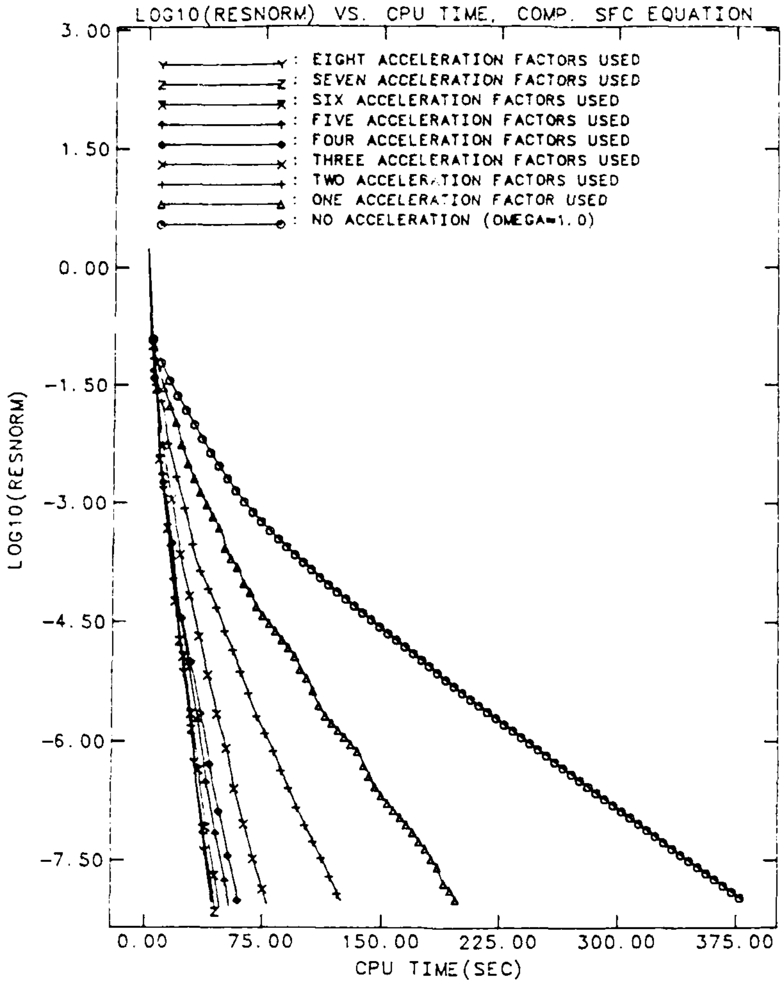


Fig. 8. Residual norm versus the computing time, compressible SFC equation.

corrections obtained from each intermediate step and by the additional arithmetic operations to determine the coefficients of the minimizing polynomial. In practice, a maximum gain in computational efficiency can be obtained with a moderate number (usually not more than five) of intermediate steps. The requirement for additional storage linearly increases with the number of intermediate time steps used and represents only a fraction of the computer storage required by the GMRES method [8].

24

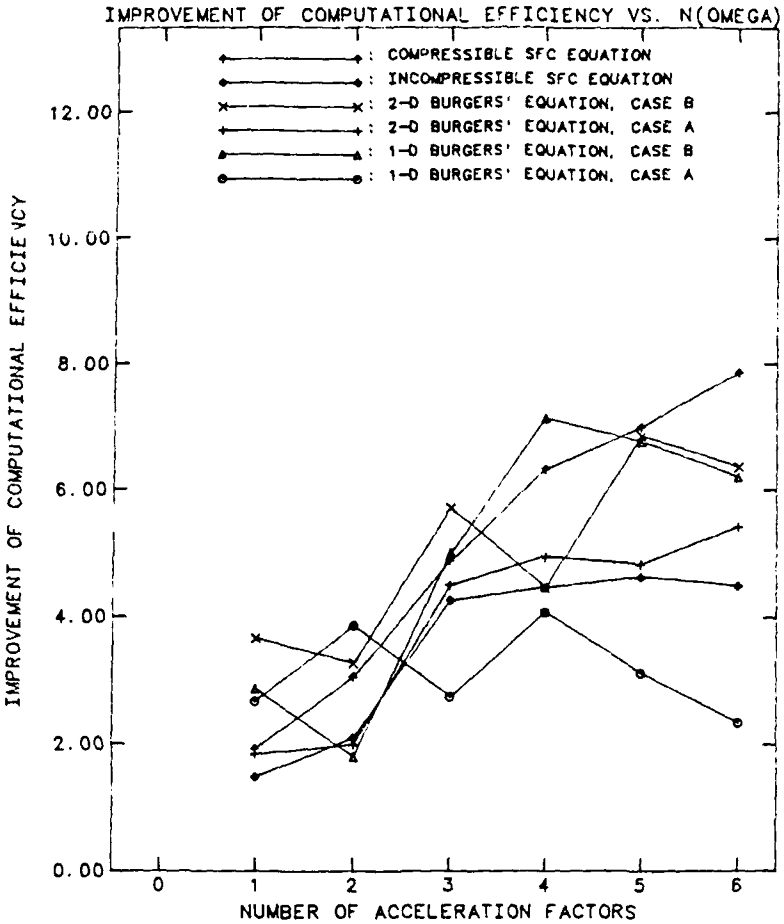


Fig. 9. Improvement of computational efficiency versus the number of acceleration factors.

References

[1] A.R. Mitchell and D.F. Griffiths, The Finite Difference Method in Partial Differential Equations (Wiley, New York, 1980).

[2] S.I. Cheng, Computational Fluid Dynamics—Numerical Method for Partial Differential Equations (Scientific Publishing Company, Peking, 1984) (in Chinese).

[3] S.R. Kennon and G.S. Dulikravich, Optimal acceleration factors for iterative solution of linear and nonlinear differential systems, Comput. Meths. Appl. Mech. Engr. 47 (1984) 357–367.

[4] S.R. Kennon, Optimal acceleration factors for iterative solution of linear and nonlinear differential systems, AIAA Paper 85-0162, Reno, NV, 1985.

[5] C.-Y. Huang, S.R. Kennon and G.S. Dulikravich, Generalized non-linear minimal residual (GNLMR) method for iterative algorithms, Comput. Appl. Math. 16 (1986) 215–232.

[6] C.-Y. Huang and G.S. Dulikravich, Stream function and stream-function-coordinate (SFC) formulation for inviscid flow field calculations, Comput. Meths. Appl. Mech. Engrg. 59 (1986) 155–177.

[7] H. Ratschek and J. Rokne, Computer Methods for the Range of Functions (Ellis Horwood, Chichester, U.K., 1984).

[8] Y. Saad and M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, Research Rept. YALEU/DCS/RR-254, Yale University, New Haven, CT, 1983.

[9] L.B. Wigton, N.J. Yu and D.P. Young, GMRES acceleration of computational fluid dynamics codes, AIAA Paper 85-1494, 1985.

[10] K.N. Ghia, U. Ghia and C.T. Shin, Adaptive grid generation for flows with local high gradient regions, ASME FED-5, Houston, TX (1983) 35–4[illegible].

25

[2] S.I. Cheng, Computational Fluid Dynamics—Numerical Method for Partial Differential Equations (Scientific Publishing Company, Peking, 1984) (in Chinese).
[3] S.R. Kennon and G.S. Dulikravich, Optimal acceleration factors for iterative solution of linear and nonlinear differential systems, Comput. Meths. Appl. Mech. Engr. 47 (1984) 357–367.
[4] S.R. Kennon, Optimal acceleration factors for iterative solution of linear and nonlinear differential systems, AIAA Paper 85-0162, Reno, NV, 1985.
[5] C.-Y. Huang, S.R. Kennon and G.S. Dulikravich, Generalized non-linear minimal residual (GNLMR) method for iterative algorithms, Comput. Appl. Math. 16 (1986) 215–232.
[6] C.-Y. Huang and G.S. Dulikravich, Stream function and stream-function-coordinate (SFC) formulation for inviscid flow field calculations, Comput. Meths. Appl. Mech. Engrg. 59 (1986) 155–177.
[7] H. Ratschek and J. Rokne, Computer Methods for the Range of Functions (Ellis Horwood, Chichester, U.K., 1984).
[8] Y. Saad and M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, Research Rept. YALEU DCS RR-254, Yale University, New Haven, CT, 1983.
[9] L.B. Wigton, N.J. Yu and D.P. Young, GMRES acceleration of computational fluid dynamics codes, AIAA Paper 85-1494, 1985.
[10] K.N. Ghia, U. Ghia and C.T. Shin, Adaptive grid generation for flows with local high gradient regions, ASME FED-5, Houston, TX (1983) 35–47.

26

Second Announcement and Registration Form

CONFERENCE ON ITERATIVE METHODS FOR LARGE LINEAR SYSTEMS

October 19,20, and 21, 1988

The University of Texas at Austin

Celebrating the 65th birthday of David M. Young, Jr.

OBJECTIVE: This conference will be dedicated to providing an overview of the state of the art in the use of iterative methods for solving sparse linear systems with an eye to contributions of the past, present, and future. The emphasis will be placed upon identifying current and future research directions in the main-stream of modern scientific computing. Recently, the use of iterative methods for solving linear systems has experienced a resurgence of activity as scientists attack extremely complicated three dimensional problems using vector and parallel supercomputers. Many research advances in the development of iterative methods for high-speed computers over the past forty years will be reviewed as well as focusing on current research.

INVITED SPEAKERS:

L. Adams (Washington University)
O. Axelsson (University of Nijmegen, Netherlands)
G. Birkhoff (Harvard University)
P. Concus (Lawrence Berkeley Laboratory)
L. Ehrlich (John Hopkins University)
H. Elman (University of Maryland)
G. Golub (Stanford University)
L. Hageman (Bettis Atomic Power Laboratory)
R. Lynch (Purdue University)
T. Manteuffel (Los Alamos National Laboratory)
J. Ortega (University of Virginia)
J. Rice (Purdue University)
P. Saylor (University of Illinois)
R. Varga (Kent State University)
W. Wachspress (University of Tennessee)
M. Wheeler (Rice University)
O. Widlund (Courant Institute)
D. Young (University of Texas)

ORGANIZATIONS: The host organization is the Center for Numerical Analysis (CNA) of The University of Texas at Austin. The members of the local organizing committee are D. Kincaid, L. Hayes, G. Carey and W. Cheney. The conference is being co-sponsored by the Special Interest Groups for Linear Algebra and Supercomputing of the Society for Industrial and Applied Mathematics.

ACCOMMODATIONS: To make reservations call the Austin Marriott at the Capital [(512) 478-1111 or (800) 228-9290] or mail the enclosed card for rooms at the special conference rate of $55 for single or double rooms. All reservations are handled on a first-come-first-served basis. Reservations must be received by October 4, 1988. Reservations made after this date are subject to guest room availability.

REGISTRATION: Conference advanced registration fee is $95 (SIAG/LA or SIAG/SC members $85), if received by September 1, 1988. If paid after this date or at the conference, the registration fee is $125. This fee includes morning and afternoon coffee breaks, a reception, a banquet honoring Professor Young, and other activities. The student registration fee is $15 which allows admission to the technical sessions and coffee breaks only. To register, detach and mail the registration form below. For additional information, contact the CNA at the address below or at Tel: (512) 471-1242 ; Arpanet: sheri@sally.utexas.edu; Bitnet: sheri@uta3081.

Mrs. Katy Burrell
Conference Secretary
Center for Numerical Analysis
RLM Bldg 13.150
University of Texas at Austin
Austin, Texas 78713-8510

(detach and mail)

Presented at the Conference on Iterative Methods For Large Linear Systems, Univ. of Texas, Austin, October 19-21, 1988

DISTRIBUTED MINIMAL RESIDUAL (DMR) METHOD FOR EXPLICIT ALGORITHMS APPLIED TO NONLINEAR SYSTEMS

Seungsoo Lee
Graduate Assistant

George S. Dulikravich
Associate Professor

Daniel J. Dorney
Graduate Assistant

Department of Aerospace Engineering
Pennsylvania State University, University Park, PA 16802

ABSTRACT

A new algorithm for the acceleration of explicit iterative schemes for the numerical solution of nonlinear systems of partial differential equations has been developed. The method is based on the idea of allowing each partial differential equation in the system to approach the converged solution at its own optimal speed. The DMR (Distributed Minimal Residual) method introduces a separate sequence of optimal weighting factors to be used for each component of the general solution vector. The acceleration scheme was applied to a highly nonlinear coupled system of four time-dependent partial differential equations of inviscid gasdynamics in conjunction with the finite volume Runge-Kutta explicit time-stepping algorithm. Using DMR without multigridding, between 30% and 70% of the total computational efforts were saved in the subsonic compressible flow calculations. DMR method offers most time savings when applied to stiff systems of equations.

Several attempts have been made to accelerate the iterative convergence of this method. They include local time stepping, implicit residual smoothing, enthalpy damping and multigrid techniques. Also, an extrapolation procedure based on the power method and the Minimal Residual Method (MRM) were applied to the finite volume Runge-Kutta method. In the MRM, a weighted combination of the corrections at consecutive iteration levels is extrapolated and the weights are chosen to minimize the L_2 norm of the future residual. The extrapolation was performed without considering the properties of the governing equations. The GNLMR (Generalized Non-Linear Minimal Residual) method utilizes the information about the governing equations. It has been applied successfully to a number of scalar nonlinear partial differential equations.

Both MRM and GNLMR method are based on using the same values of optimal weighting factors for the corrections to every equation in a system. Since each component of the solution vector in a system of equations has its own convergence speed, the sequence of optimal weights could be allowed to be different for each component. This concept is the essence of the DMR method. Thus, for example, we combined corrections from four consecutive time steps by introducing four weighting factors to each of the four equations. Hence, a set of sixteen algebraic equations needs to be solved to determine the four sequences of four weighting factors in each of them. The DMR method requires about 200% more storage than the original non-accelerated algorithm.

GEORGE S. DULIKRAVICH

FED-Vol. 66

ADVANCES AND APPLICATIONS IN COMPUTATIONAL FLUID DYNAMICS

presented at

THE WINTER ANNUAL MEETING OF
THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
CHICAGO, ILLINOIS
NOVEMBER 27–DECEMBER 2, 1988

sponsored by

THE FLUIDS ENGINEERING DIVISION, ASME

edited by

O. BAYSAL
OLD DOMINION UNIVERSITY

THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
United Engineering Center 345 East 47th Street New York, N.Y. 10017

29

ITERATIVE ACCELERATION AND PHYSICALLY BASED DISSIPATION FOR EULER EQUATIONS OF GASDYNAMICS

G. S. Dulikravich, Associate Professor
D. J. Dorney and S. Lee, Graduate Students
Department of Aerospace Engineering
The Pennsylvania State University
University Park, Pennsylvania

ABSTRACT

A new algorithm for the acceleration of explicit iterative schemes for a system of partial differential equations has been developed. The method is based on the idea of allowing each partial differential equation in the system to approach the converged solution at its own optimal speed. The DMR (Distributed Minimal Residual) method allows a separate sequence of optimal weighting factors to be used for each component of the general solution vector. The acceleration scheme was applied to the system of time-dependent Euler equations of inviscid gasdynamics in conjunction with the finite volume Runge-Kutta explicit time-stepping method with the Jameson's Artificial Dissipation (AD) terms and the newly formulated Physically Based Dissipation (PBD) model. The PBD model uses physical dissipation terms from the Navier-Stokes equations of gasdynamics, while enforcing slip boundary conditions of inviscid gasdynamics and utilizing spatially varying viscosity coefficients. Tests were performed for various flow conditions, including internal flow, flow around a cylinder and flow over an airfoil with AD and PBD. Using DMR, between 30% and 70% of the computational efforts were saved in the subsonic compressible flow calculations.

INTRODUCTION

When the Euler equations of inviscid gasdynamics are solved using a central difference scheme (e.g., a Runge-Kutta time-stepping scheme [1]), decoupling of odd and even grid points allows oscillations to develop which cause instabilities in the numerical algorithm. These oscillations can be damped by either explicitly or implicitly adding a certain amount of artificial dissipation [2].

Contemporary artificial dissipation models for central difference schemes usually consist of an ad hoc combination of second order and fourth order artificial (non-physical) dissipation terms [2]. The second order terms are used to damp oscillations in shock regions, while the fourth order terms ensure monotonic convergence to steady state in smooth flow regions [1,3,4].

Most of the existing artificial dissipation formulations are intuitive [5,1]. The intuitive formulations generate artificial dispersion terms [4], which are partially neutralized by adding higher order artificial dissipation terms. Only after a trial-and-error process can it be found that the coefficients multiplying second and fourth order artificial dissipation which are appropriate for very low speeds are orders of magnitude smaller then the coefficients that are appropriate for transonic speeds. It has been shown that using different amounts of second order and fourth order dissipation can produce different numerical results that are often misleading, especially in the case of transonic shocked flows with separation [6,7].

It can be concluded that the intuitive formulations for artificial dissipation which have been favored in the past are only marginally reliable. Their accuracy is still an open question [6,7,8,9,10] since there is no known exact solution to the Euler equations for a shocked flow with inviscid separation. Thus, the existing artificial dissipation models are subject to constant modifications [4,11,12,13] in order to meet the requirements posed by different flow speed regimes.

One objective of this paper is to introduce physically consistent [14] model for the dissipation to be used in the numerical solution of Euler and Navier-Stokes equations. The other objective is to introduce a new concept for convergence acceleration.

Several attempts have been made in the past to accelerate the iterative convergence of the Runge-Kutta method [15]. They include local time stepping [1], implicit residual smoothing [1], enthalpy damping [1] and multigrid techniques [12,16]. Also, the extrapolation procedure based on the power method and the Minimal Residual Method (MRM) were applied [16] to the Runge-Kutta method. In the MRM [16], a weighted combination of the corrections at consecutive iteration levels are extrapolated and the weights are chosen to minimize the L_2 norm of the future residual. Since the extrapolation was performed without considering the properties of the governing equations, it may upset the solution procedure. The GNLMR (Generalized Non-Linear Minimal

30

Residual) method [17,18,19,20] used the information from the governing equations. It has been applied successfully to a number of single nonlinear partial differential equations including the Euler equations.

Both MRM and GNLMR method use the same sequence of optimal weights for the corrections to every equation in a system. Since each component of the solution vector has its own convergence speed, the sequence of optimal weights should be allowed to vary from component to component. Thus, the objective of this paper is to present the theory constituting the Distributed Minimal Residual DMR method and todemonstrate the advantages of the new algorithm with a number of computational examples.

EULER EQUATIONS OF GASDYNAMICS

The two-dimensional Euler equations in conservative form and cartesian coordinates can be written as

$$\bar{Q}_t + \bar{E}_x + \bar{F}_y = 0 \tag{1}$$

Here, the subscripts t, x, y represent partial derivatives with respect to time, and to x, y coordinates, respectively. The general vectors

$\bar{Q}$, $\bar{E}$, and $\bar{F}$ are defined as

$$\bar{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_o \end{bmatrix} \quad \bar{E} = \begin{bmatrix} \rho u \\ \rho u^2+p \\ \rho uv \\ \rho uh_o \end{bmatrix} \quad \bar{F} = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2+p \\ \rho vh_o \end{bmatrix} \tag{2}$$

where ρ, p, u, v, e_o and h_o are the non-dimensional values of local density, thermodynamic pressure, x-component of velocity, y-component of velocity, total mass-specific energy and total mass-specific enthalpy, respectively.

Equation of state for a calorically perfect gas can be expressed as

$$p = (\gamma-1)\left(\rho e_o - \frac{1}{2}\left(\frac{(\rho u)^2}{\rho} + \frac{(\rho v)^2}{\rho}\right)\right) \tag{3}$$

where γ represents the ratio of specific heats. The total mass-specific enthalpy, h_o, is defined as

$$h_o = e_o + \frac{p}{\rho} \tag{4}$$

For the analysis of flows about arbitrary geometries, the formulation can be generalized by using, say, fixed body-fitted non-orthogonal coordinates ξ and η, so that

$$\xi = \xi(x,y) \ ; \ \eta = \eta(x,y) \tag{5}$$

Thus, in the computational (ξ,η) domain, the two-dimensional Euler equations in a strongly conservative form become

$$Q_t + E_\xi + F_\eta = 0 \tag{6}$$

where

$$Q = \frac{1}{D}\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_o \end{bmatrix} \quad E = \frac{1}{D}\begin{bmatrix} \rho U \\ \rho uU+p\xi_x \\ \rho vU+p\xi_y \\ \rho h_o U \end{bmatrix} \quad F = \frac{1}{D}\begin{bmatrix} \rho V \\ \rho uV + p\eta_x \\ \rho vV + p\eta_y \\ \rho h_o V \end{bmatrix} \tag{7}$$

Here

$$D = \det\left(\frac{\partial(\xi,\eta)}{\partial(x,y)}\right) = \frac{1}{\det\left(\frac{\partial(x,y)}{\partial(\xi,\eta)}\right)} = 1/\begin{vmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{vmatrix} \tag{8}$$

Thus

$$\xi_x/D = y_\eta; \quad \eta_x/D = -y_\xi \ ; \ \xi_y/D = -x_\eta; \ \eta_y/D = x_\xi \tag{9}$$

The contravariant components U,V of the velocity vector are related to the cartesian components u,v as follows

$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tag{10}$$

EXISTING ARTIFICIAL DISSIPATION MODEL

A typical stage of the multistage Runge-Kutta [15] time-stepping scheme for the Euler equations [1,4] is

$$Q^{(n)} = Q^{(o)} + \alpha_n \Delta t \left(E_\xi^{(n-1)} + F_\eta^{(n-1)} + P\right) \tag{11}$$

where $P_{i,j}$ is the artificial dissipation [1,4] given as

$$P_{i,j} = \left(P^\xi_{i+1/2,j} - P^\xi_{i-1/2,j}\right) + \left(P^\eta_{i,j+1/2} - P^\eta_{i,j-1/2}\right) \tag{12}$$

Calculation of the artificially dissipative terms is done similarly [1,4] for all conservation laws. For example,

$$P^\xi_{i+\delta/2,j} = \varepsilon^{(2)}_{i+\delta/2,j}\, Q^\xi_{i+\delta/2,j} - \varepsilon^{(4)}_{i+\delta/2,j}\left(\theta_{i+\delta,j}\, Q^{\xi\xi}_{i+\delta,j} - \theta_{i,j}\, Q^{\xi\xi}_{i,j}\right) \tag{13}$$

$$P^\eta_{i,j+\delta/2} = \varepsilon^{(2)}_{i,j+\delta/2}\, Q^\eta_{i,j+\delta/2} - \varepsilon^{(4)}_{i,j+\delta/2}\left(\theta_{i,j+\delta}\, Q^{\eta\eta}_{i,j+\delta} - \theta_{i,j}\, Q^{\eta\eta}_{i,j}\right) \tag{14}$$

where $\delta = \pm 1$. The remaining terms are defined as follows:

$$\theta_{i,j} = \frac{1}{(D\,\Delta t^*)_{i,j}} \tag{15}$$

$$Q^\xi_{i+\delta/2,j} = Q_{i+\delta,j} - Q_{i,j} \tag{16}$$

$$Q^{\xi\xi}_{i+\delta,j} = Q_{i+2\delta,j} - 2Q_{i+\delta,j} + Q_{i,j} \tag{17}$$

$$Q^{\xi\xi}_{i,j} = Q_{i+1,j} - 2Q_{i,j} + Q_{i-1,j} \tag{18}$$

31

with similar expressions for the other terms of this type. The coefficients of second and fourth order artificial dissipation are defined [4,1], respectively, as

$$\epsilon^{(2)}_{i+\delta/2,j} = \theta_{i,j}\ \kappa^{(2)} \max\left(\nu^{\xi\xi}_{i,j}\ ;\ \nu^{\xi\xi}_{i+\delta,j}\right) \tag{19}$$

$$\epsilon^{(4)}_{i+\delta/2,j} = \max\left(0;\ \kappa^{(4)} - \kappa^{(2)} \max\left(\nu^{\xi\xi}_{i,j}\ ;\ \nu^{\xi\xi}_{i+\delta,j}\right)\right) \tag{20}$$

with similar expressions for the other terms of this type, where $\kappa^{(2)} = 1/4$ to $1/2$ and $\kappa^{(4)} = 1/128$ to $1/64$ are typical constants [3]. Here, the local "directional pressure sensor" is defined as

$$\nu^{\xi\xi}_{i,j} = \left| \frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{p_{i+1,j} + 2p_{i,j} + p_{i-1,j}} \right| \tag{21}$$

Similarly

$$\nu^{\eta\eta}_{i,j} = \left| \frac{p_{i,j+1} - 2p_{i,j} + p_{i,j-1}}{p_{i,j+1} + 2p_{i,j} + p_{i,j-1}} \right| \tag{22}$$

PHYSICALLY BASED DISSIPATION (PBD) MODEL

Instead of using an intuitive non-physical formulation for the artificial dissipation, we suggest that the dissipation should be based on actual physical dissipation, that is, it should be physically consistent. We propose that to solve the Euler equations of inviscid flow, one should actually solve the complete Navier-Stokes equations of viscous and heat conducting flow subject to perfect slip boundary conditions and spatially varying coefficients of viscosity [14]. Thus, the PBD model represents a physically consistent formulation since the Euler equations of inviscid gasdynamics represent an extreme case of Navier-Stokes equations when the physical dissipation becomes negligible.

The Navier-Stokes equations of unsteady, viscous, laminar flow allowing for heat conduction (assuming Fourier's law) expressed in non-dimensional form and non-orthogonal curvilinear coordinates can be summarized as

$$Q_t + E_\xi + F_\eta = \frac{1}{Re}\left(E^v_\xi + F^v_\eta\right) \tag{23}$$

where R_e is the Reynolds number and E^v_ξ, F^v_η incorporate physically dissipative terms due to shear viscosity, secondary viscosity and heat conductivity. The generalized viscous flux vectors are

$$E^v = \frac{1}{D}\begin{bmatrix} 0 \\ E_1 \\ E_2 \\ E_3 \end{bmatrix}\ ;\quad F^v = \frac{1}{D}\begin{bmatrix} 0 \\ F_1 \\ F_2 \\ F_3 \end{bmatrix} \tag{24}$$

where

$$\begin{bmatrix} E_1 & E_2 \\ F_1 & F_2 \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \begin{bmatrix} \tau_{xx} & \tau_{xy} \\ \tau_{yx} & \tau_{yy} \end{bmatrix} \tag{25}$$

and

$$\begin{bmatrix} E_3 \\ F_3 \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \begin{bmatrix} \tau_{xx} & \tau_{xy} \\ \tau_{yx} & \tau_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} - \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \begin{bmatrix} q_x \\ q_y \end{bmatrix} \tag{26}$$

Here, the components of the non-dimensional viscous stress tensor expressed in terms of ξ,η coordinates are:

$$\tau_{xx} = \mu''\left(\xi_x u_\xi + \eta_x u_\eta\right) + \lambda\left(\xi_y v_\xi + \eta_y v_\eta\right) \tag{27}$$

$$\tau_{xy} = \mu''\left(\xi_y u_\xi + \eta_y u_\eta + \xi_x v_\xi + \eta_x v_\eta\right) \tag{28}$$

and the non-dimensional heat conduction flux is

$$q_x = \frac{\mu}{(\gamma-1)M_\infty^2 Pr}\left(\xi_x T_\xi + \eta_x T_\eta\right) \tag{29}$$

Here, $\mu'' = 2\mu + \lambda$ is the longitudinal viscosity coefficient, M_∞ is the Mach number of the uniform flow at infinity, Pr is the Prandtl number and T is the absolute temperature. Since Rankine-Hugoniot shock jump conditions are possible only [21] if Stokes hypothesis ($\lambda/\mu = -2/3$) is enforced, we use this relation in actual computations.

In the PBD formulation, the shear viscosity coefficient, μ, is forced to vary throughout the flowfield by means of an appropriate "sensor". The physical thermodynamic pressure, p, appears in the equations of gasdynamics in the form of its first derivative. Consequently, we have decided to use the pressure sensor which is based on the streamwise first derivative of the pressure, that is,

$$\mu_{i,j} = \frac{C\mu}{(u^2 + v^2)^{1/2}}\left(Up_\xi + Vp_\eta\right) = C\, p_s \mu \tag{30}$$

Here, C is a user specified constant. Using numerical experimentation, we have found that $10 < C < 20$ for the range of freestream Mach numbers $0.1 < M_\infty < 30$. We have experimented with a number of different "sensors" and found that the three-point average streamwise first derivative of pressure gives a robust scheme

$$\mu_{i,j} = \left(\mu_{i+1,j} + \mu_{i,j} + \mu_{i-1,j}\right) / 3 \tag{31}$$

Obviously, this is just one among many possible suggestions for the "sensor." Other choices might be, for example,

sensor based on divergence:

$$\mu_{i,j} = C\,|\nabla \cdot \vec{V}|\,\mu \tag{32}$$

sensor based on Mach number:

$$\mu_{i,j} = \mu C\left(M_\xi U + M_\eta V\right)/|\vec{V}| \tag{33}$$

32

sensor based on M^2:

$$\mu_{i,j} = \mu C(M_\xi^2 U + M_\eta^2 V)/|\vec{V}| \tag{34}$$

DISTRIBUTED MINIMAL RESIDUAL (DMR) METHOD

Global residual of the finite volume method at time level t can be expressed in two dimensions as

$$r^t = \iint \frac{\partial Q}{\partial t}\, dS = -\iint \left(\frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta}\right) dS \tag{35}$$

where S is the surface area of the single grid cell and components Q, E and F of the generalized solution vector are defined in Eq. 2. In the case of, say, four-step explicit Runge-Kutta algorithm one needs four intermediate time steps to advance the solution from the global time level (t) to (t+1).

We plan to use corrections from M previous consecutive time levels to update the value of Q to (t+1) global time level. Thus,

$$Q^{t+1} = Q^t + \sum_m^M \Omega^m \tag{36}$$

where

$$\Omega^m = \begin{bmatrix} \omega_1^m \ \Delta_1^m \\ \omega_2^m \ \Delta_2^m \\ \vdots\ M \ \vdots\ M \\ \omega_L \ \Delta_L \end{bmatrix} \tag{37}$$

and Δ_ℓ^m are the corrections for each of the $\ell = 1,\dots,L$ equations in the system (Eq. 2) at each of the m=1,...,M global time steps. Therefore, substituting Eq. 36 in Eq. 35, the new local residual for the single cell will be

$$r^{t+1} = -\iint \left[\frac{\partial}{\partial \xi} E\,(Q^t + \sum_m^M \Omega^m) + \frac{\partial}{\partial \eta} F\,(Q^t + \sum_m^M \Omega^m)\right] dS \tag{38}$$

Using a Taylor series expansion truncated after the first term results in

$$r^{t+1} = r^t - \sum_m^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\Omega^m\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\Omega^m\right)\right] dS \tag{39}$$

Define the global residual R^t as a sum of the squares of the local residuals, that is,

$$R^t = \sum_i^I \sum_j^J (r^t)^* (r^t) \tag{40}$$

where I and J define the grid size and the superscript * designates the transpose. Then, the global residual at the next global time level will be

$$R^{t+1} = \sum_i^I \sum_j^J \left\{r^t - \sum_n^M \iint\left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\Omega^n\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\Omega^n\right)\right] dS\right\}^*$$

$$\cdot \left\{r^t - \sum_m^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\Omega^m\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\Omega^m\right)\right] dS\right\} \tag{41}$$

To minimize R^{t+1}, it is necessary to use the values of ω_ℓ that satisfy

$$\frac{\partial R^{t+1}}{\partial \omega_\ell^m} = 0 \tag{42}$$

for all m and ℓ. Thus, from Eq. 41 and Eq. 42 it follows that

$$\sum_i^I \sum_j^J (r^t)^* \left\{\iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS\right\}$$

$$= \sum_i^I \sum_j^J \sum_n^M \left\{\iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\Omega^n\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\Omega^n\right)\right] dS\right\}^* \tag{43}$$

$$\cdot \left\{\iint\left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS\right\}$$

where $\frac{\partial \Omega^m}{\partial \omega_\ell^m} = \{\Delta_k^m \delta_{k\ell}\}$ and $\delta_{k\ell}$ is the Kronecker delta.

Notice that

$$\frac{\partial E}{\partial Q}\Omega^n = \sum_q^L \omega_q^n \frac{\partial E}{\partial Q}\frac{\partial \Omega^n}{\partial \omega_q^n} \tag{44}$$

and

$$\frac{\partial F}{\partial Q}\Omega^n = \sum_q^L \omega_q^n \frac{\partial F}{\partial Q}\frac{\partial \Omega^n}{\partial \omega_q^n} \tag{45}$$

Let

$$A_\ell^m = \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS \tag{46}$$

Note that A_ℓ^m is not a function of ω's. Then, Eq. (43) becomes

$$\sum_i^I \sum_j^J (r^t)^* A_\ell^m = \sum_i^I \sum_j^J \sum_n^M \sum_q^L \omega_q^n (A_q^n)^* A_\ell^m \tag{47}$$

Let

$$C_{q\ell}^{nm} = \sum_i^I \sum_j^J (A_q^n)^* A_\ell^m \tag{48}$$

and

$$B_\ell^m = \sum_i^I \sum_j^J (r^t)^* A_\ell^m \tag{49}$$

Then

$$\sum_n^M \sum_q^L \omega_q^n C_{q\ell}^{nm} = B_\ell^m \tag{50}$$

or

$$\sum_n^M (\omega_1^n C_{1\ell}^{nm} + \omega_2^n C_{2\ell}^{nm} + \omega_3^n C_{3\ell}^{nm} + \dots + \omega_L^n C_{L\ell}^{nm}) = B_\ell^m \tag{51}$$

resulting in a system of LxM equations for the unknown distributed optimal acceleration factors ω_ℓ^m. In the case of two-dimesional Euler equations, L = 4. Thus, we must solve the following system of 4xM equations in order to determine the 4xM optimal values of ω_ℓ^m.

$$
\left[\begin{array}{cccc|ccc}
C_{11}^{11} & C_{21}^{11} & C_{31}^{11} & C_{41}^{11} & C_{11}^{21} & \cdots & C_{41}^{M1} \\
C_{12}^{11} & C_{22}^{11} & C_{32}^{11} & C_{42}^{11} & C_{12}^{21} & \cdots & C_{42}^{M1} \\
C_{13}^{11} & C_{23}^{11} & C_{33}^{11} & C_{43}^{11} & C_{13}^{21} & \cdots & C_{43}^{M1} \\
C_{14}^{11} & C_{24}^{11} & C_{34}^{11} & C_{44}^{11} & C_{14}^{21} & \cdots & C_{44}^{M1} \\
\hline
C_{11}^{21} & C_{21}^{21} & C_{31}^{21} & C_{41}^{21} & C_{11}^{21} & \cdots & C_{41}^{M1} \\
\vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\
C_{14}^{1M} & C_{24}^{1M} & C_{34}^{1M} & C_{44}^{1M} & C_{14}^{2M} & \cdots & C_{44}^{M4}
\end{array}\right]
\cdot
\left[\begin{array}{c}
\omega_1^1 \\ \omega_2^1 \\ \omega_3^1 \\ \omega_4^1 \\ \hline \omega_1^2 \\ \vdots \\ \omega_4^M
\end{array}\right]
=
\left[\begin{array}{c}
B_1^1 \\ B_2^1 \\ B_3^1 \\ B_4^1 \\ \hline B_1^2 \\ \vdots \\ B_4^M
\end{array}\right]
\qquad (52)
$$

We have decided to use M = 4, that is, four consecutive global time steps.

Thus, four sequences of four optimal values of ω were used in Eq. 36 and Eq. 37 to update the solution to the next global time level.

RESULTS

The PBD and DMR concepts were applied to three model test cases: external flow around a cylinder, internal channel flow past a 10% circular arc, and external flow around a NACA 0012 airfoil.

Figure 1 shows the 65x33 O-type computational grid around a cylinder. Figures 2 and 3 show the convergence histories using the existing Artificial Dissipation (AD) model and the Physically Based Dissipation (PBD) model with and without the application of DMR with $M_\infty = 0.2$. Using DMR, the number of iterations needed to achieve the same level of residual is reduced by almost 50%. The savings in computational time is about 50% (Figures 4 and 5) using the AD and the PBD model with the application of DMR at $M_\infty = 0.2$. The savings in cpu time can be seen in Table 1, which presents the run times and residuals for several of the test cases. In Table 1, Res0 is the starting residual and Res is the final residual.

Figures 6 and 7 show convergence histories using the AD and the PBD with and without the DMR model for the $M_\infty = 0.4$. This is a clear indication that the present formulation of DMR is incapable of CPU time savings as the locally sonic flow conditions are approached.

At $M_\infty = 0.1$ the compressible Euler equations become a stiff system of partial differential equations. Figure 8 shows convergence history using the AD model with and without DMR. The large CPU time savings demonstrate the ability of DMR to treat stiff systems of equations (Figure 9).

Figure 10 illustrates the 65x17 H-type channel grid with a 10% circular arc bump on the floor. Figures 11-12 show that at $M_\infty = 0.5$ using the AD or the PBD model with the DMR yields about 50% savings in CPU time. Figures 13 and 14 show that using AD or the PBD model with DMR at $M_\infty = 0.6$ saves less than 50% in CPU time. The pressure contours using the AD and the PBD models were identical at $M_\infty = 0.6$ (Figure 15).

Figure 16 shows the 65x33 C-type clustered grid around a NACA 0012 airfoil. From Figure 17 it appears that using the AD model and DMR at $M_\infty = 0.63$ does not accelerate convergence. From Figure 18, the it is clear that using the PBD model allows the DMR to perform better even for this transonic shocked flow case resulting in over 30% saving in the CPU.

Finally, the PBD model was compared to the AD model by applying them to lifting and nonlifting transonic flows. The 129x33 C-type grid around a NACA 0012 airfoil is shown in Figure 19. Figure 20 shows isobars using the AD model at $M_\infty = 0.8$, $\alpha = 0.0°$. Figure 21 shows isobars when using the PBD model at $M_\infty = 0.8$, $\alpha = 1.25°$. Again, the PBD model yields a sharp shock.

CONCLUSIONS

A new physically based dissipation model has been presented. Advantages of the new model include:

1. The second order dissipation used in the PBD model represents actual physically consistent dissipation from the Navier-Stokes equations for compressible, viscous, heat conducting fluid flow.
2. The PBD model does not contaminate the continuity equation.
3. The PBD formulation maintains high accuracy. Actually, for flows with stronger shocks, the PBD formulation gives results comparable to TVD schemes.
4. The PBD concept can be applied to Navier-Stokes equations, too. The higher order physically consistent dissipation terms can be based on dissipation due to radiation heat transfer and heat generation due to chemical reactions.
5. An Euler solver with the PBD formulation easily converts to a Navier-Stokes solver by fixing the value of viscosity coefficient and by specifying no-slip boundary conditions on solid surfaces.

A conceptually new method termed Distributed Minimal Residual (DMR) has been developed and successfully applied to the acceleration of an explicit iterative algorithm for the numerical solution of a nonlinear system of Euler equations governing inviscid gasdynamics. The main idea of using a separate sequence of optimal acceleration factors for each of the equations in the system was theoretically formulated a numerically proven on a number of test cases. This means that the partial differential equations governing mass, x-momentum, y-momentum and energy conservation were accelerated according to their own separate optimal sequences of acceleration factors that have a common objective of minimizing the global residual of the entire system at each consecutive integration time step. DMR in its present form works best for low Mach number flows when the Euler equations become exceedingly stiff.

ACKNOWLEDGEMENTS

This work was supported by the Air Force Office of Scientific Research Numerical Mathematics Program under the supervision of Dr. John P. Thomas and Dr. Arje Nachman. The manuscript was typed by Ms. Amy Myers.

REFERENCES

1. Jameson, A., Schmidt, W. and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper No. 81-1259, Palo Alto, CA, June, 1981.

2. Pulliam, T. H., "Artificial Dissipation Models for the Euler Equations," AIAA Journal, Vol. 24, No. 12, December 1986, pp. 1931-1940.

3. Chima, R.V., Turkel, E., and Schaffer, S., "Comparison of Three Explicit Multigrid Methods for the Euler and Navier-Stokes Equations," NASA TM 88878, 1986; also ICOMP-86-3.

4. Caughey, D. A. and Turkel, E., "Effects of Numerical Dissipation on Finite-Volume Solutions of Compressible Flow Problems," AIAA paper 88-0621, Reno, NV, January 1988.

5. MacCormack, R. W. and Baldwin, B. S., "A Numerical Method for Solving the Navier-Stokes Equations With Application to Shock-Boundary Layer Interaction," AIAA paper 75-1, January 1975.

6. Salas, M. D., "Recent Developments in Transonic Euler Flow Over a Circular Cylinder," presented at the 10th IMACS World Congress on Systems Simulation and Scientific Computation, Montreal, Canada, August 8-13, 1982.

7. Abdy, G. L., "Inviscid Separation at Transonic Speeds," M.Sc. Thesis, Dept. of Mechanical and Aerospace Eng., Arizona State Univ., Tempe, AZ, December 1986.

8. Atkins, H. L., "Numerical Comparison of Time-Accurate Algorithms for a Nonlinear Shock-Propagation Problem," AIAA paper 85-1663.

9. Pandolfi, M. and Larocca, F., "Transonic Flow About a Circular Cylinder," proceedings of the conference in celebration of 60th birthday of Prof. G. Moretti, Aug. 1987.

10. Hughes, T., "Calculations of 2-D Euler Flows With a New Petrov-Galerkin Finite Element Method," Notes on Num. Fluid Mechanics, Vieweg-Verlag.

11. Raj, P. and Brennan, J., "Improvements to an Euler Aerodynamics Method for Transonic Flow Analysis," AIAA paper 87-0040, January 1987.

12. Chima, R. V., "Analysis of Inviscid and Viscous Flows in Cascades with an Explicit Multiple-Grid Algorithm," NASA TM 83636, 1983.

13. Jameson, A. and Mavriplis, D., "Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh," AIAA Journal, Vol. 24, No. 4, April 1986, pp. 611-618.

14. Dulikravich, G. S., Dorney, D. J., and Lee, S. S., "A Physically Based Dissipation (PBD) Model for the Euler Equations of Gasdynamics," submitted for publication.

15. Wambecq, A., "Rational Runge-Kutta Methods for Solving Systems of Ordinary Differential Equations," Computing, Vol. 20, 1978, pp. 333-342.

16. Hafez, M., Parlette, E., and Salas, M. D., "Convergence Acceleration of Iterative Solutions of Euler Equations for Transonic Flow Computations," AIAA Paper 85-1641, July 1985.

17. Kennon, S. R., and Dulikravich, G. S., "Optimal Acceleration Factors for Iterative Solution of Linear and Nonlinear Differential Systems," Computer Methods in Applied Mechanics and Engineering, 47, 1984, pp. 357-367.

18. Kennon, S. R., "Optimal Acceleration Factors for Iterative Solution of Linear and Nonlinear Differential Systems," AIAA Paper 85-0162, January 1985, Reno, Nevada.

19. Huang, C. Y., Kennon, S. R., and Dulikravich, G. S., "Generalized Nonlinear Minimal Residual (GNLMR) Method for Iterative Algorithms," Journal of Computational and Applied Mathematics, 16, 1986, pp. 215-232.

20. Huang, C. Y. and Dulikravich, G. S., "Fast Iterative Algorithms Based on Optimized Explicit Time-Stepping," Computer Methods in Applied Mechanics and Engineering, 63, August 1987, pp. 15-36.

21. Dulikravich, G. S., Mortara, K. W. and Marraffa, L., "A Physically Consistent Model for Artificial Dissipation in Transonic Potential Flow Computations," AIAA paper 88-3654 presented at the 1st National Congress of Fluid Dynamics, Cincinnati, OH, July 1988.

ARTIFICIAL DISSIPATION METHOD

Cylinder

M	No DNLMR	DNLMR	Res	Res0
0.1	1778.12	735.63	-3.0	-0.7441330
0.2	1774.30	837.39	-2.8	-0.4233154
0.3	2094.92	1251.21	-3.7	-0.2269995
0.4	2063.57	1835.53	-4.5	-8.1262559E-02

Airfoil

M	No DNLMR	DNLMR	Res	Res0
0.63	1499.51	1735.21	-2.9	1.009068

PHYSICALLY BASED DISSIPATION METHOD

Cylinder

M	No DNLMR	DNLMR	Res	Res0
0.2	2212.95	1594.33	-2.4	-0.4341858
0.3	1986.46	1548.81	-2.7	-0.2433041
0.4	2442.75	2164.490	-3.0	-0.1030241

Airfoil

M	No DNLMR	DNLMR	Res	Res0
0.63	2612.46	1529.06	-0.78	0.9581382

Table 1.

Comparison of CPU time (sec) for Artificial Dissipation (AD) and Physically Based Dissipation (PBD)

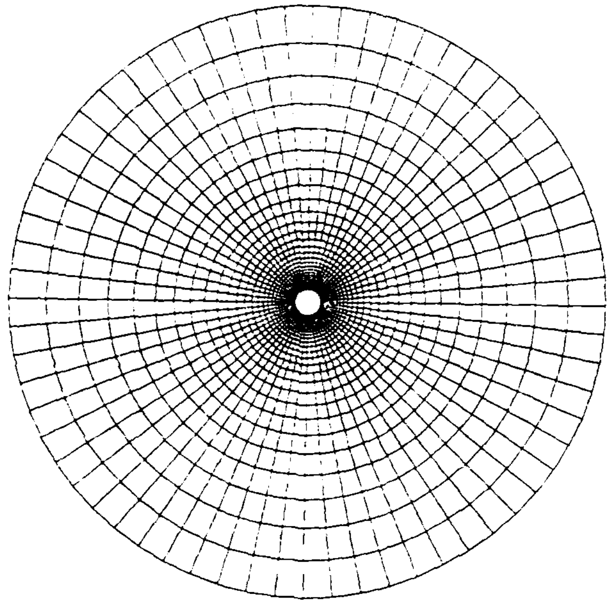


Figure 1.

Circle: Computational O-type grid (65 x 33)

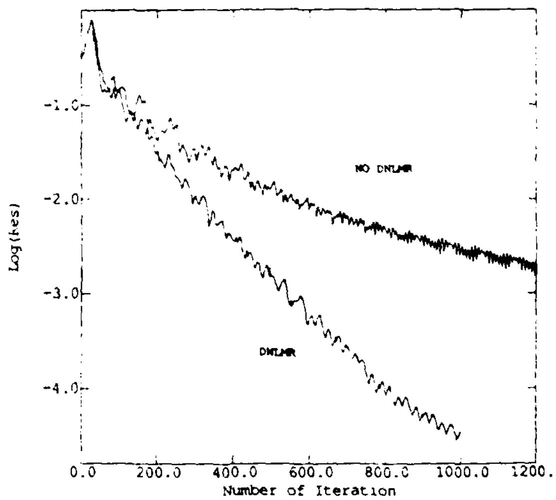


Figure 2.

Circle: Convergence history; AD; $M_\infty = 0.2$.

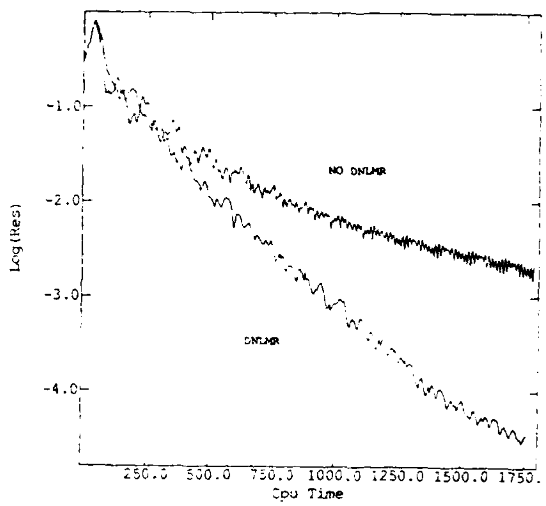


Figure 3.

Circle: Convergence history; PBD; $M_\infty = 0.2$.

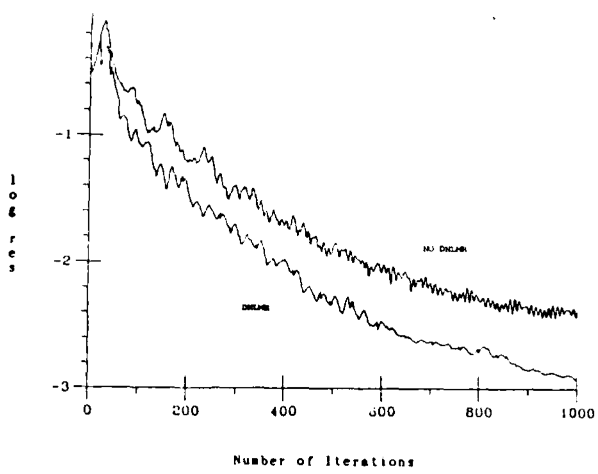


Figure 4.

Circle: CPU history; AD; $M_\infty = 0.2$.

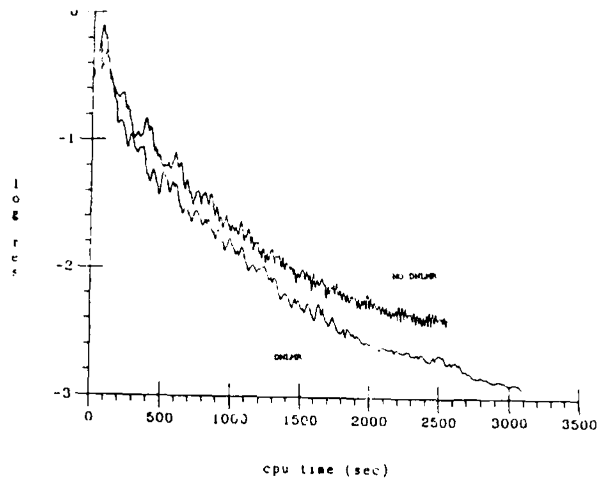


Figure 5.

Circle: CPU history; PBD; $M_\infty = 0.2$.

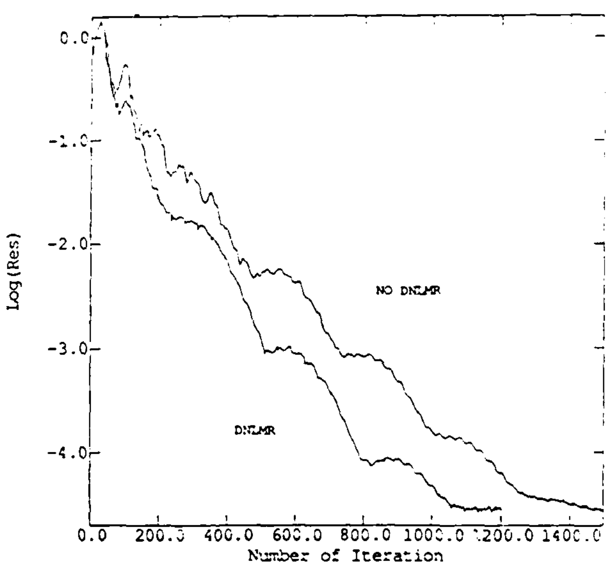


Figure 6.

Circle: Convergence history; AD; $M_\infty = 0.4$.

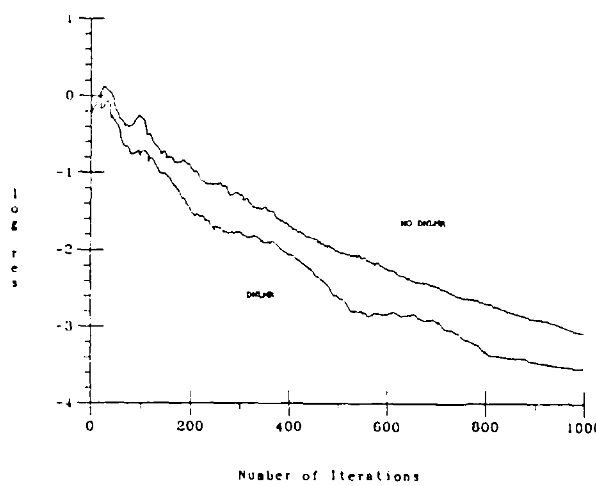


Figure 7.

Circle: Convergence history; PBD; $M_\infty = 0.4$.

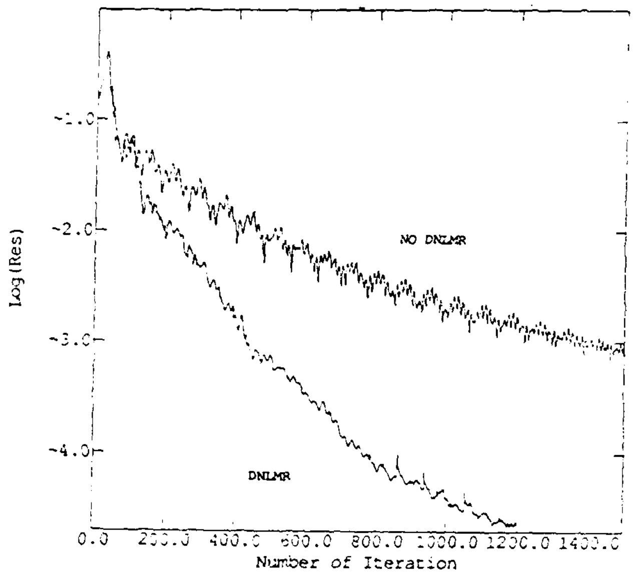


Figure 8.

Circle: Convergence history; AD; $M_\infty = 0.1$.

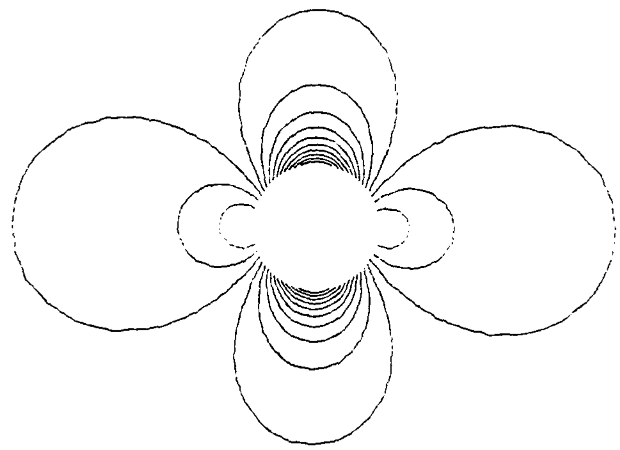


Figure 9.

Circle: Isobars; AD; $M_\infty = 0.1$.

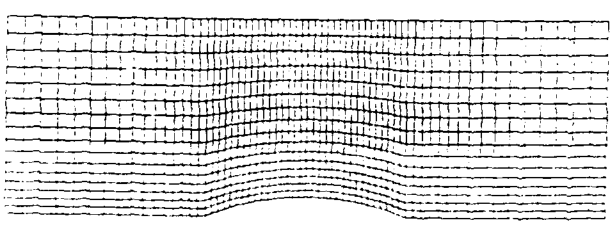


Figure 10.

Channel: Computational grid (65 x 17)

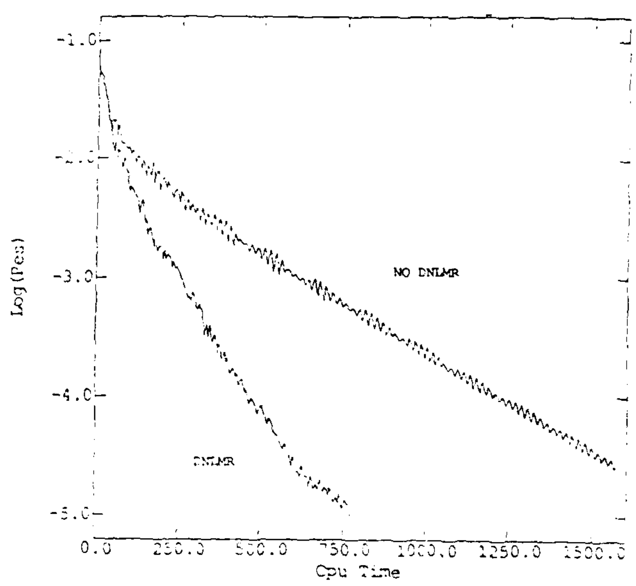


Figure 11.

Channel: CPU history; AD; $M_\infty = 0.5$.

38

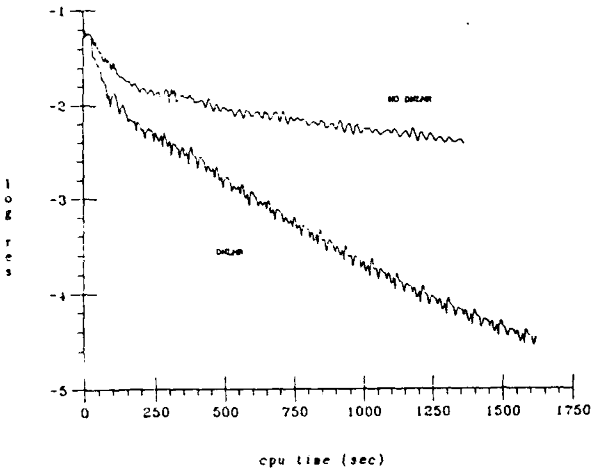


Figure 12.

Channel: CPU history; PBD; $M_\infty = 0.5$.

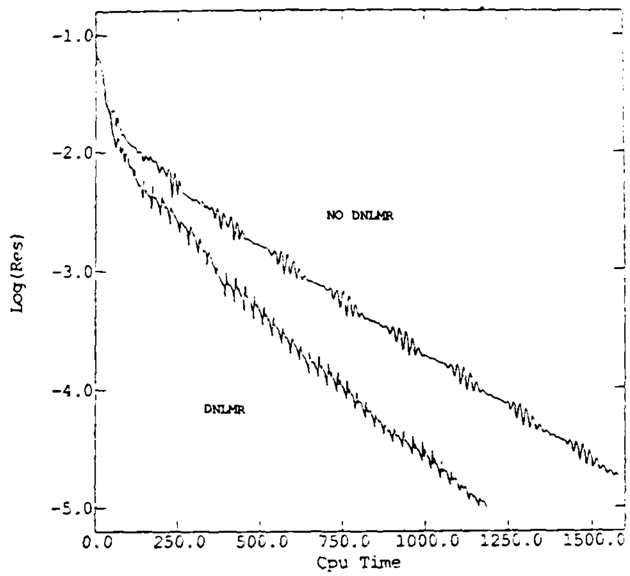


Figure 13.

Channel: CPU history; AD; $M_\infty = 0.6$.

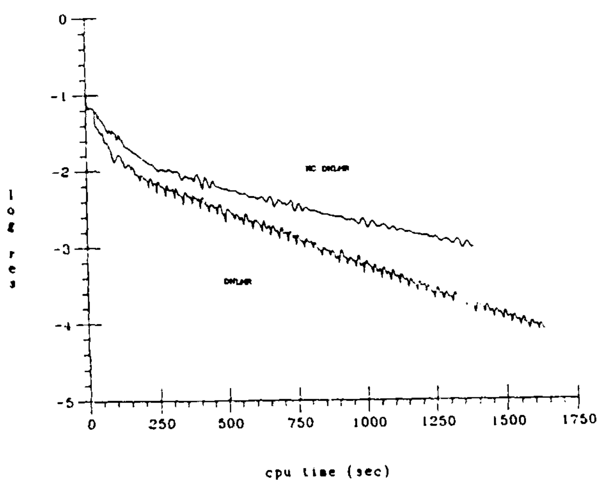


Figure 14.

Channel: CPU history; PBD; $M_\infty = 0.6$.

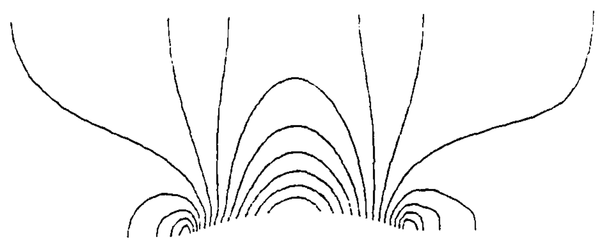


Figure 15.

Channel: Isobars; AD & PBD; $M_\infty = 0.6$.

39

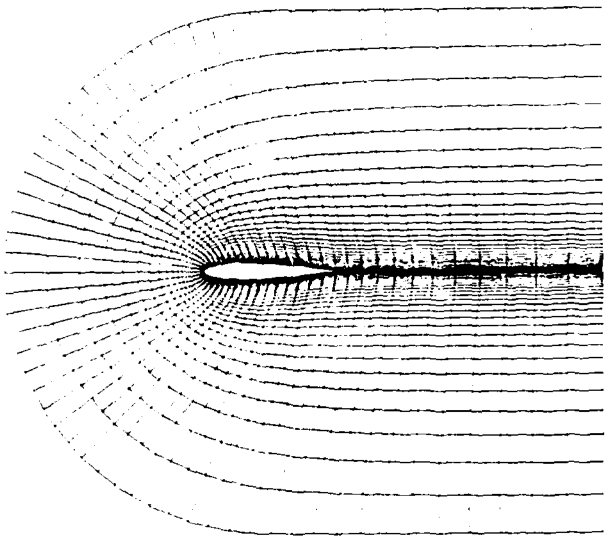


Figure 16.

NACA0012: Computational grid (65 x 33)

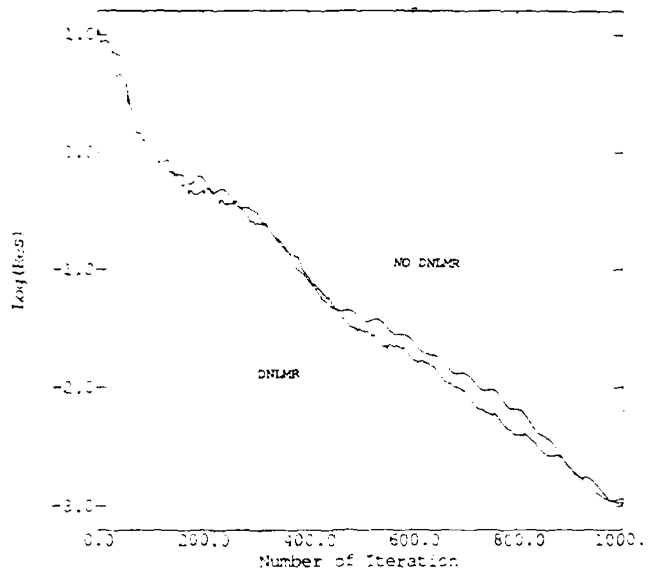


Figure 17.

NACA0012: Convergence history; AD; $M_\infty = 0.63$.

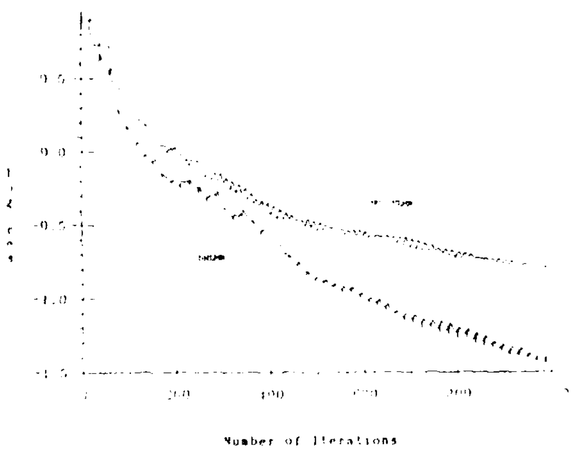


Figure 18.

NACA0012: Convergence history; PBD; $M_\infty = 0.63$.

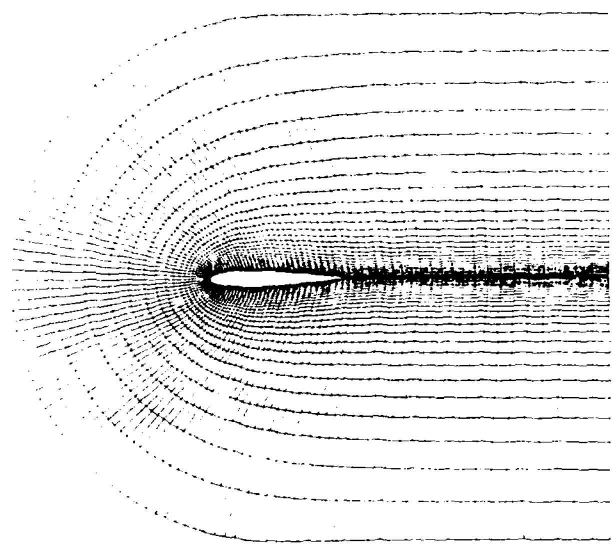


Figure 19.

NACA0012: Computational grid (129 x 33)

40

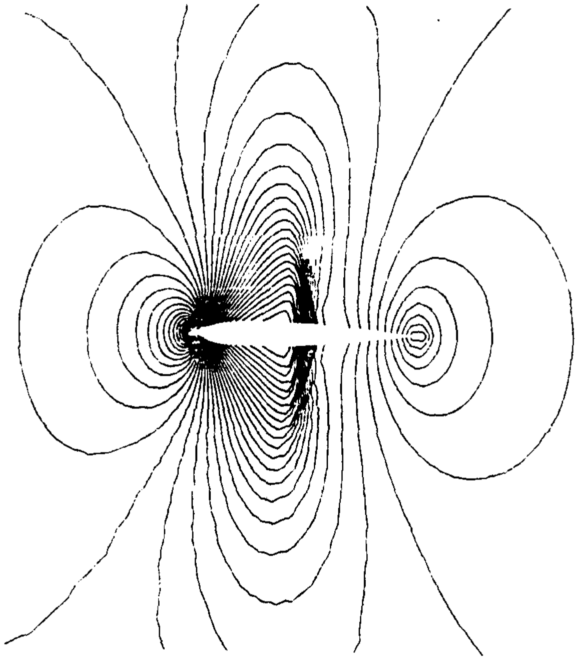


Figure 20.

NACA0012: Isobars; AD; $M_\infty = 0.8; \alpha = 0.0°$.

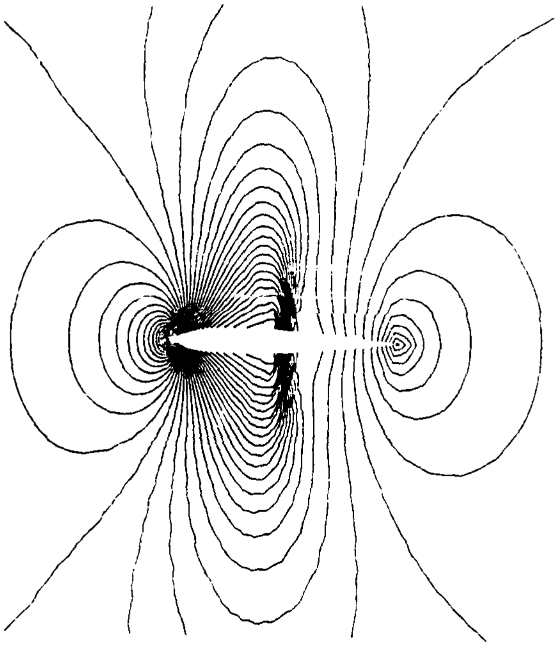


Figure 21.

NACA0012: Isobars; PBD; $M_\infty = 0.8; \alpha = 0.0°$.

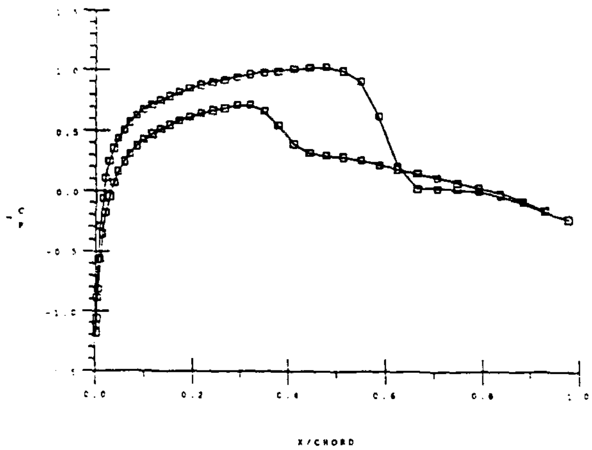


Figure 22.

NACA0012: Surface pressure coefficients; AD; $M_\infty = 0.8; \alpha = 1.25°$.

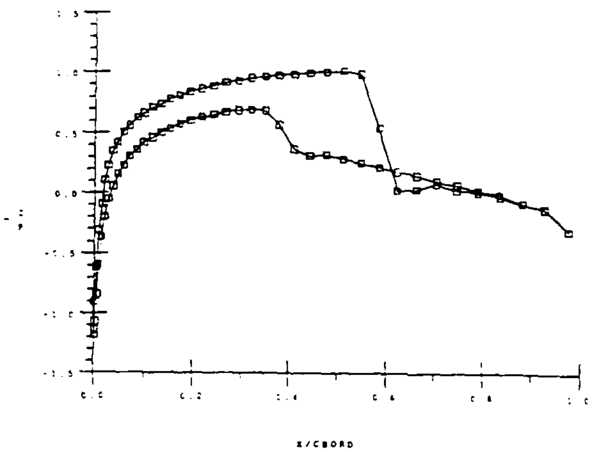


Figure 23.

NACA0012: Surface pressure coefficients; PBD; $M_\infty = 0.8; \alpha = 1.25°$.

AIAA'89

AIAA-89-0097
Acceleration of Iterative Algorithms for Euler Equations of Gasdynamics

Seungsoo Lee, George S. Dulikravich
and Daniel J. Dorney

The Pennsylvania State University
University Park, PA 16802

27th Aerospace Science Meeting
January 9-12, 1989
Reno, Nevada

42

ACCELERATION OF ITERATIVE ALGORITHMS
FOR EULER EQUATIONS OF GASDYNAMICS

Seungsoo Lee, Graduate Assistant
George S. Dulikravich, Associate Professor
Daniel J. Dorney, Graduate Assistant

Department of Aerospace Engineering
Pennsylvania State University, University Park, PA 16802

ABSTRACT

A new algorithm for the acceleration of iterative schemes for the numerical solution of systems of partial differential equations has been developed. The method is based on the idea of allowing each partial differential equation in the system to approach the converged solution at its own optimal speed. The DMR (Distributed Minimal Residual) method allows a separate sequence of optimal weighting factors to be used for each equation in the system. The acceleration scheme was applied to the system of time-dependent Euler equations of inviscid gasdynamics in conjunction with the finite volume Rational Runge-Kutta (RRK) explicit time-stepping algorithm. Using DMR without multigridding, between 30% and 70% of the total computational efforts were saved in the subsonic compressible flow calculations. DMR method in its present version seems to be especially suitable for stiff systems of equations. It required less than double amount of storage of the original non-accelerated algorithm.

INTRODUCTION

One of the successful, explicit methods used to solve Euler and Navier-Stokes equations governing compressible flows subject to the various flow conditions is the Rational Runge-Kutta (RRK) time-stepping algorithm [1,2]. It is based on the finite volume technique with 2nd-4th order blended non-physical (artificial) dissipation [1]. Several attempts have been made to accelerate the iterative convergence of this method. They include local time stepping [1], implicit residual smoothing [1], enthalpy damping [1] and multigrid techniques [3]. Also, an extrapolation procedure based on the power method and the Minimal Residual Method (MRM) were applied [3] to the finite volume Runge-Kutta method together with multigridding. In the MRM [3], a weighted combination of the corrections at consecutive iteration levels is extrapolated and the weights are chosen to minimize the L_2 norm of the future residual. The extrapolation was performed without considering the specific properties of the governing equations. The GNLMR (Generalized Non-Linear Minimal Residual) method [4,5,6,7] utilizes the information from the governing equations. It has been applied successfully to a number of scalar linear and nonlinear partial differential equations.

Both MRM and GNLMR method use the same values of optimal weights for the corrections to every equation in a system. Nevertheless, since each component of the solution vector in a system of equations has its own convergence speed, the sequence of optimal weights could be allowed to vary from equation to equation. The authors believe that this concept underlying the Distributed Minimal Residual (DMR) method is similar to the general idea behind the preconditioning techniques. With the *preconditioning*, the eigenvalues of the system are changed so that the different CFL (Courant-Friedrichs-Levy) number can be used for each characteristic variable. This paper presents the theory constituting the DMR method and demonstrates the advantages of the new algorithm with a number of computational examples. Applications of the DMR to the system of Euler equations of inviscid gasdynamics are presented. The formulation can be equally well applied to other systems of differential equations and to other types of numerical integration algorithms.

TIME-DEPENDENT EULER EQUATIONS OF INVISCID GASDYNAMICS

The system of time-dependent Euler equations of gasdynamics in two-dimensional space can be written in a general conservative form as

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta} = 0 \qquad (1)$$

where the global solution vectors combining mass, x-momentum, y-momentum and energy conservation equations are defined as

$$Q = \frac{1}{D}\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_o \end{bmatrix} \qquad E = \frac{1}{D}\begin{bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ \rho(e_o + p)U \end{bmatrix}$$

43

$$F = \frac{1}{D}\begin{bmatrix} \rho V \\ \rho u V + n_x p \\ \rho v V + n_y p \\ \rho(e_o + p)V \end{bmatrix} \qquad (2)$$

Here, ρ, u, v, p, e_o are the density, x and y components of the velocity vector, thermodynamic pressure, and mass-specific total energy, respectively. In addition, U, V, ξ, η and D are the contravariant velocity vector components, non-orthogonal curvilinear computational coordinates, and determinant of the Jacobian transformation $\partial(\xi,\eta)/\partial(x,y)$, respectively.

The contravariant components U and V of the velocity vector in the body-conforming (ξ,η) coordinate system are given by

$$U = \xi_x u + \xi_y v \qquad (3)$$

$$V = \eta_x u + \eta_y v \qquad (4)$$

The total energy per unit mass for a calorically perfect gas is

$$e_o = c_v T + \frac{1}{2}(u^2+v^2) \qquad (5)$$

where c_v is the specific heat at constant volume and T is the absolute temperature. The determinant of the Jacobian geometric transformation matrix is

$$D = \xi_x \eta_y - \xi_y \eta_x \qquad (6)$$

FINITE VOLUME RUNGE-KUTTA TIME-STEPPING ALGORITHM

In the *finite volume method* [1], the governing equations are integrated over each computational cell in the (ξ,η) computational plane. With the help of the divergence theorem, the surface integral is transformed into a sum of line integrals. These integrals are discretized with the assumption that the fluxes are constant along the cell faces. Each quantity at the cell face is evaluated as the average of the values at the neighboring cell centers (cell centered scheme).

The cell centered finite volume method is identical to the central difference scheme on a uniform grid. It is known that the central difference scheme produces odd-even decoupling. To suppress this tendency,the artificial dissipation terms are added to the discretized equation [1]. The mixture of 2nd and 4th order artificial dissipation terms [1] was used.

$$dQ = d_\xi Q + d_\eta Q \qquad (7)$$

where d is the artificial dissipation operator and Q is the vector defined in Eq. 2. The two terms on the right hand side of Eq. 7 are contributions from the two computational directions. They can be written [1] as:

$$d_\xi Q = d_{i+1/2j} - d_{i-1/2j}; \quad d_\eta Q = d_{ij+1/2} - d_{ij-1/2} \qquad (8)$$

The terms on the right hand sides of Eq. 8 are similar [1]. For example,

$$d_{i+1/2j} = \frac{1}{D\,\Delta t}\{\epsilon^{(2)}_{i+1/2j}(Q_{i+1j} - Q_{ij}) - \epsilon^{(4)}_{i+1/2j}(Q_{i+2j} - 3Q_{i+1j} + 3Q_{ij} - Q_{i-1j})\} \qquad (9)$$

where the second and fourth order coefficients multiplying the flux derivative terms are flow adaptive coefficients. The scaling with the area D and the local time step, Δt, is included [8] to correspond to the formulation of the Euler equations in the transformed plane. A pressure sensor is introduced to locate regions requiring large amounts of artificial dissipation. It is based on the second derivative of pressure [1,9]

$$\nu_{ij} = \frac{|p_{i+1j} - 2p_{ij} + p_{i-1j}|}{p_{i+1j} + 2p_{ij} + p_{i-1j}} \qquad (10)$$

The flow adaptive coefficients are then calculated [1] as:

$$\epsilon^{(2)}_{i+1/2j} = k^{(2)}\max(\nu_{i+1j}, \nu_{ij})$$

$$k^{(2)} = 1/4$$

$$\epsilon^{(4)}_{i+1/2j} = \max(0, (k^{(4)} - \epsilon^{(2)}_{i+1/2j}))$$

$$k^{(4)} = 1/256 \qquad (11)$$

The system of time-dependent Euler equations is known to be of hyperbolic type and the boundary conditions should be applied according to the direction of the characteristics. At the inflow and outflow boundaries, the incoming Riemann invariant is specified and the outgoing Riemann invariant is extrapolated from the interior points. Also, the entropy and the tangential velocity are prescribed at the inflow. At the outflow, these quantities are extrapolated from the interior of the domain.

At the solid wall, the normal momentum equation is used to evaluate the wall pressure. The contravariant velocity component U at the ghost cells inside the solid body is extrapolated, while the contravariant velocity component V is reflected from the wall.

An explicit Runge-Kutta time-stepping [2,1] scheme is used to evolve the solution in time. The 4th order Runge-Kutta scheme is given by

$$Q^{(0)} = Q^n \qquad (12)$$

$$Q^{(1)} = Q^n - \frac{\Delta t}{4}(NQ^{(0)} - dQ^{(0)}) \qquad (13)$$

$$Q^{(2)} = Q^n - \frac{\Delta t}{3}(NQ^{(1)} - dQ^{(0)}) \qquad (14)$$

$$Q^{(3)} = Q^n - \frac{\Delta t}{2}(NQ^{(2)} - dQ^{(0)}) \qquad (15)$$

$$Q^{(4)} = Q^n - \Delta t\,(NQ^{(3)} - dQ^{(0)}) \qquad (16)$$

$$Q^{n+1} = Q^{(4)} \qquad (17)$$

44

where N is the discretization operator of the finite volume method. The artificial dissipation is evaluated at the beginning of each Runge-Kutta global step and it was not updated during the intermediate steps. Linear stability analysis indicates that the explicit Runge-Kutta scheme is stable if CFL ≤ 2.8. If the grid spacing in (ξ,η) plane is uniform $\Delta\xi = \Delta\eta = 1$, then the time step is given [9] by

$$\Delta t_\xi = \frac{1}{|U| + a\,(\xi_x^{\,2} + \xi_y^{\,2})^{1/2}} \tag{18}$$

$$\Delta t_\eta = \frac{1}{|V| + a\,(\eta_x^{\,2} + \eta_y^{\,2})^{1/2}} \tag{19}$$

where a is the local speed of sound and the combined time step [9] is

$$\Delta t = \left(\frac{\Delta t_\xi\,\Delta t_\eta}{\Delta t_\xi + \Delta t_\eta}\right) \mathrm{CFL} \tag{20}$$

DISTRIBUTED MINIMAL RESIDUAL METHOD (DMR)

Local residual of the finite volume method at time level t can be expressed as

$$r^t = \iint \frac{\partial Q}{\partial t}\, dS = -\iint \left(\frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta}\right) dS \tag{21}$$

where S is the surface of the single grid cell and components Q, E and F of the generalized solution vector are defined in Eq. 2.

We plan to use corrections from M consecutive time levels to update the value of Q to (t+1) global time level. Thus,

$$Q^{t+1} = Q^t + \sum_m^M \Omega^m \tag{22}$$

where

$$\Omega^m = \begin{bmatrix} \omega_1^{\,m}\,\Delta_1^{\,m} \\ \omega_2^{\,m}\,\Delta_2^{\,m} \\ \cdots\cdots \\ \omega_L^{\,m}\,\Delta_L^{\,m} \end{bmatrix} \tag{23}$$

and Δ_ℓ^m are the corrections and ω_ℓ^m are the weights for each of the $\ell=1,\dots,L$ equations in the system (Eq. 2) at each of the $m=1,\dots,M$ consecutive global time levels. Therefore, upon substituting Eq. 22 in Eq. 21, the new local residual for the single cell will be

$$r^{t+1} = -\iint \left[\frac{\partial}{\partial \xi} E\left(Q^t + \sum_m^M \Omega^m\right) + \frac{\partial}{\partial \eta} F\left(Q^t + \sum_m^M \Omega^m\right)\right] dS \tag{24}$$

Using a Taylor series expansion truncated after the first term results in

$$r^{t+1} = -\iint \left[\frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta}\right] dS - \sum_m^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\Omega^m\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\Omega^m\right)\right] dS \tag{25}$$

or

$$r^{t+1} = r^t - \sum_m^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\Omega^m\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\Omega^m\right)\right] dS \tag{26}$$

Define the global residual R^t as a sum of the squares of the local residuals, that is,

$$R^t = \sum_i^I \sum_j^J (r^t)^* \,(r^t) \tag{27}$$

where I and J define the grid size and the superscript * designates the transpose of an array. Then, the global residual at the next global time level will be

$$R^{t+1} = \sum_i^I \sum_j^J \left\{r^t - \sum_n^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\Omega^n\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\Omega^n\right)\right] dS\right\}^* \cdot \left\{r^t - \sum_m^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\Omega^m\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\Omega^m\right)\right] dS\right\} \tag{28}$$

To minimize R^{t+1}, it is necessary to use the values of ω_ℓ^m that satisfy

$$\frac{\partial R^{t+1}}{\partial \omega_\ell^m} = 0 \tag{29}$$

for all m and ℓ. Thus, from Eq. 29 and Eq. 28 it follows that

$$\sum_i^I \sum_j^J \left\{r^t - \sum_n^M \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\Omega^n\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\Omega^n\right)\right] dS\right\}^* \cdot \left\{\iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS\right\} = 0 \tag{30}$$

or

$$\sum_i^I \sum_j^J (r^t)^* \left\{\iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS\right\} = \sum_i^I \sum_j^J \sum_n^M \left\{\iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\Omega^n\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\Omega^n\right)\right] dS\right\}^* \cdot \left\{\iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\,\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\,\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS\right\} \tag{31}$$

45

where $\frac{\partial \Omega^m}{\partial \omega_\ell^m} = \{\Delta_k^m \delta_{k\ell}\}$ and $\delta_{k\ell}$ is the Kronecker delta.

$$(32)$$

Notice that

$$\frac{\partial E}{\partial Q}\Omega^n = \sum_q^L \omega_q^n \frac{\partial E}{\partial Q}\frac{\partial \Omega^n}{\partial \omega_q^n} \tag{33}$$

and

$$\frac{\partial F}{\partial Q}\Omega^n = \sum_q^L \omega_q^n \frac{\partial F}{\partial Q}\frac{\partial \Omega^n}{\partial \omega_q^n} \tag{34}$$

Let

$$A_\ell^m = \iint \left[\frac{\partial}{\partial \xi}\left(\frac{\partial E}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F}{\partial Q}\frac{\partial \Omega^m}{\partial \omega_\ell^m}\right)\right] dS \tag{35}$$

Note that A_ℓ^m is not a function of ω's. Then, Eq. (31) becomes

$$\sum_i^I \sum_j^J (r^t)^* A_\ell^m = \sum_i^I \sum_j^J \sum_n^M \sum_q^L \omega_q^n (A_q^n)^* A_\ell^m \tag{36}$$

Let

$$C_{q\ell}^{nm} = \sum_i^I \sum_j^J (A_q^n)^* A_\ell^m \tag{37}$$

and

$$B_\ell^m = \sum_i^I \sum_j^J (r^t)^* A_\ell^m \tag{38}$$

Then

$$\sum_n^M \sum_q^L \omega_q^n C_{q\ell}^{nm} = B_\ell^m \tag{39}$$

or

$$\sum^M (\omega_1^n C_{1\ell}^{nm} + \omega_2^n C_{2\ell}^{nm} + \omega_3^n C_{3\ell}^{nm} + \ldots\ldots + \omega_L^n C_{L\ell}^{nm}) = B_\ell^m \tag{40}$$

resulting in a system of LxM equations for the LxM unknown optimal acceleration factors ω_ℓ^m. In the case of two-dimensional Euler equations, L = 4. Thus, we must solve simultaneously the following system of 4xM equations in order to determine the 4xM optimal values of ω_ℓ^m.

$$\left[\begin{array}{cccc|ccc}
C_{11}^{11} & C_{21}^{11} & C_{31}^{11} & C_{41}^{11} & C_{11}^{21} & \ldots. & C_{41}^{M1} \\
C_{12}^{11} & C_{22}^{11} & C_{32}^{11} & C_{42}^{11} & C_{12}^{21} & \ldots. & C_{42}^{M1} \\
C_{13}^{11} & C_{23}^{11} & C_{33}^{11} & C_{43}^{11} & C_{13}^{21} & \ldots. & C_{43}^{M1} \\
C_{14}^{11} & C_{24}^{11} & C_{34}^{11} & C_{44}^{11} & C_{14}^{21} & \ldots. & C_{44}^{M1} \\
\hline
C_{11}^{12} & C_{21}^{12} & C_{31}^{12} & C_{41}^{12} & C_{11}^{22} & \ldots. & C_{41}^{M2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
C_{14}^{1M} & C_{24}^{1M} & C_{34}^{1M} & C_{44}^{1M} & C_{14}^{2M} & \ldots. & C_{44}^{M4}
\end{array}\right]
\cdot
\left[\begin{array}{c}
\omega_1^1 \\ \omega_2^1 \\ \omega_3^1 \\ \omega_4^1 \\ \hline \omega_1^2 \\ \vdots \\ \omega_4^M
\end{array}\right]
=
\left[\begin{array}{c}
B_1^1 \\ B_2^1 \\ B_3^1 \\ B_4^1 \\ \hline B_1^2 \\ \vdots \\ B_4^M
\end{array}\right] \tag{41}$$

We have decided to combine four consecutive time steps (M=4). Since the twodimensional Euler equations form a system that has four equations (L = 4), these four sequences of four optimal values of ω can then be used in Eq. 23 and Eq. 22 to update the solution to the next global time level t+1.

The matrices $\partial E/\partial Q$ and $\partial F/\partial Q$ that are needed for evaluation of the coefficients in the above matrix are given as:

$$
\frac{\partial E}{\partial Q} = \begin{bmatrix} 0 & \xi_x & \xi_y & 0 \\ \frac{\gamma-1}{2}\xi_x(u^2+v^2) - uU & (2-\gamma)\xi_x u + U & \xi_y u - (\gamma-1)\xi_x v & (\gamma-1)\xi_x \\ \frac{\gamma-1}{2}\xi_y(u^2+v^2) - vU & \xi_x v - (\gamma-1)\xi_y u & (2-\gamma)\xi_y v + U & (\gamma-1)\xi_y \\ [(\gamma-1)(u^2+v^2) - \gamma e]U & h\xi_x - (\gamma-1)uU & h\xi_y - (\gamma-1)vU & \gamma U \end{bmatrix} \quad (42)
$$

$$
\frac{\partial F}{\partial Q} = \begin{bmatrix} 0 & \eta_x & \eta_y & 0 \\ \frac{\gamma-1}{2}\eta_x(u^2+v^2) - uV & (2-\gamma)\eta_x u + V & \eta_y u - (\gamma-1)\eta_x v & (\gamma-1)\eta_x \\ \frac{\gamma-1}{2}\eta_y(u^2+v^2) - vV & \eta_x v - (\gamma-1)\eta_y u & (2-\gamma)\eta_y v + V & (\gamma-1)\eta_y \\ [(\gamma-1)(u^2+v^2) - \gamma e]V & h\eta_x - (\gamma-1)uV & h\eta_y - (\gamma-1)vV & \gamma V \end{bmatrix} \quad (43)
$$

where h is the specific enthalpy per unit mass and γ in the ratio of specific heats for a calorically perfect gas.

In addition to the computer memory required by the original non-accelerated scheme [1], additional memory is needed to implement the DMR. If the grid points are IxJ and we use M global consecutive time levels to update the solution, then for the two-dimensional problem the extra memory requirement is approximately L x (2+M) x (I-2) x (J-2) and for the three-dimensional Euler equations the extra memory requirement is approximately L x (3+M) x (I-2) x (J-2) x (K-2). In the two-dimensional case this represents approximately 150% increase and in the three-dimensional case this represents approximately 175% increase in memory requirement over the original non-accelerated [1] algorithm.

Three different methods were tested for the boundary conditions on the residuals in the integrals of Eq. 35. The first method was to set the residuals at the ghost cells to be zero. The second method calculates the residuals at the ghost cells from the boundary conditions. The third method extrapolates the residuals from the interior of the flowfield. It was found that the third method gives the best results.

RESULTS

All computations were performed on a VAX 11/8550 computer in a single precision mode. The first sequence of tests of DMR was performed on the internal two-dimensional (L=4) flow problems by combining four consecutive global time steps (M=4). This means that a 16x16 matrix (Eq. 41) needs to be inverted. Figure 1 shows the computational grid for a 10% thick circular bump in a two dimensional channel. The grid size is 65x17 points. The calculations were started with uniform flow and the DMR was applied once after every 30 iterations. Figure 1 shows the convergence histories of subsonic flow calculations with $M_\infty = 0.5$. The number of iterations needed to achieve the same level of residual is reduced almost to 60%. The convergence with the DMR shows smaller oscillations than that of the original [1] scheme. It is expected that this behavior continues to the machine accuracy. The saving in computational time is about 50% for this test case.

The constant pressure contour plots of the entire flow field for both non-accelerated finite volume RRK scheme and DMR accelerated finite volume RRK scheme are shown in Fig. 4 and Fig. 5, respectively. The difference between the two is not discernable in these contour plots thus confirming that DMR method does not adversely influence the quality of the solution.

Results of the second test case are presented in Figs. 6,7,8 and 9. The entire flow field is subsonic with $M_\infty = 0.55$. For this test case, the saving was almost 40% in CPU time. It is noticeable that the convergence history shows more oscillatory behavior than for the case with $M = 0.5$. Another subsonic ($M_\infty = 0.6$) test case was tested and the results are shown in Figs. 10, 11, 12 and 13 demonstrating that a considerable amount of computation effort was saved.

Figs. 14 and 15 show the convergence histories for the transonic shocked flow case with $M_\infty = 0.675$ which is less than the flow choking Mach number of this channel. Results indicate that with the DMR, the convergence rate is not improved.

Similar trends were observed when solving Euler equations for a flow around a circle. An C-type grid consisting of 64x32 grid cells was used. For a moderately compressible subsonic flow ($M_\infty = 0.3$), DMR saves (Figs. 18 and 19) approximately 45% of CPU time. It generates results (Fig. 20) that are practically indistinguishable from the non-accelerated scheme. When the critical free stream Mach number $M_\infty = 0.4$ was used, Fig. 21 indicates and Fig. 22 confirms that the DMR method in its present form offers practically no gain when compared with the non-accelerated algorithm although the computed surface Mach numbers (Fig. 23) are equally accurate. Thus, both Ni's bump case and circle case indicate that DMR method in its present formulation offers no advantages at transonic speeds. On the other hand, the system of Euler equations becomes stiff as the Mach number decreases, thus rapidly reducing the convergence rate of the non-accelerated scheme. When using $M_\infty = 0.1$ (an almost incompressible flow), Figs. 24 and 25 demonstrate that DMR offers over 70% savings in the CPU time over the non-accelerated scheme. Fig. 26 indicates difference in the computed surface C_p values after 1200 iterations.

47

In order to account for the different local characteristic behavior of the transonic flow, it should be possible to use different sets of weights for different regions of the flowfield. Also, the artificial dissipation terms could be incorporated in the formulation of the DMR. In addition, the optimal frequency of applying the DMR needs to be investigated. In the present investigation, DMR was applied by combining four consecutive time steps after every thirty time steps.

Notice that all numerical results were obtained without the standard acceleration techniques such as explicit and implicit residual smoothing, enthalpy damping, multigridding and vectorization. These methods could be added to further accelerate the algorithm.

CONCLUSIONS

A conceptually new method termed Distributed Minimal Residual (DMR) has been developed and successfully applied to the acceleration of an explicit finite volume iterative algorithm for the numerical solution of a nonlinear system of Euler equations governing inviscid gasdynamics. The main idea of using a separate sequence of optimal acceleration factors for each of the equations in the system was theoretically formulated a numerically demonstrated with a number of examples. This means that the partial differential equations governing mass, x-momentum, y-momentum and energy conservation were accelerated according to their own separate optimal sequences of acceleration factors that have a common objective of minimizing the global residual of the entire system at each global time level. The method seems to offer significant time savings especially for stiff systems of differential equations.

ACKNOWLEDGEMENTS

This work was supported by the Air Force Office of Scientific Research/Numerical Mathematics Program under the Grant AFOSR-87-0121 supervised by Dr. John P. Thomas, Dr. Arje Nachman and Lt. Col. David Nelson. The manuscript was typed by Ms. Amy Myers. The graphics was produced on the Apple Computer system donated to our Interdisciplinary Computational Fluid Dynamics group.

REFERENCES

1. Jameson, A., Schmidt, W. and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper No. 81-1259, Palo Alto, CA, June, 1981.

2. Wambecq, A., "Rational Runge-Kutta Methods for Solving Systems of Ordinary Differential Equations," Computing, Vol. 20, 1978, pp. 333-342.

3. Hafez, M., Parlette, E., and Salas, M. D., "Convergence Acceleration of Iterative Solutions of Euler Equations for Transonic Flow Computations," AIAA Paper 85-1641, July 1985.

4. Kennon, S. R., and Dulikravich, G. S., "Optimal Acceleration Factors for Iterative Solution of Linear and Nonlinear Differential Systems," Computer Methods in Applied Mechanics and Engineering, 47, 1984, pp. 357-367.

5. Kennon, S. R., "Optimal Acceleration Factors for Iterative Solution of Linear and Nonlinear Differential Systems," AIAA Paper 85-0162, January 1985, Reno, Nevada.

6. Huang, C. Y., Kennon, S. R., and Dulikravich, G. S., "Generalized Nonlinear Minimal Residual (GNLMR) Method for Iterative Algorithms," Journal of Computational and Applied Mathematics, 16, 1986, pp. 215-232.

7. Huang, C. Y., and Dulikravich, G. S., "Fast Iterative Algorithms Based on Optimized Explicit Time-Stepping," Computer Methods in Applied Mechanics and Engineering, 63, August 1987, pp. 15-36.

8. Caughey, D. A. and Turkel, E., "Effects of Numerical Dissipation on Finite-Volume Solutions of Compressible Flow Problems, AIAA paper 86-0621, Reno, NV, January 1988.

9. MacCormack, R. W. and Baldwin, B. S. "A Numerical Method for Solving the Navier-Stokes Equations With Application to Shock-Boundary Layer Interaction," AIAA paper 75-1, January 1975.

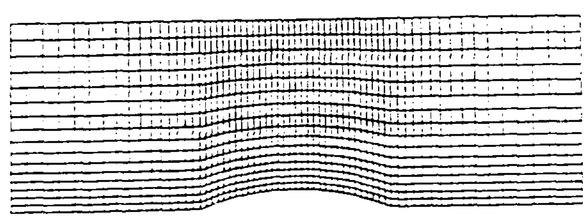


Figure 1. Computational grid for a 10% thick circular arc airfoil on the bottom wall of a straight two-dimensional channel (Ni's bump).

48

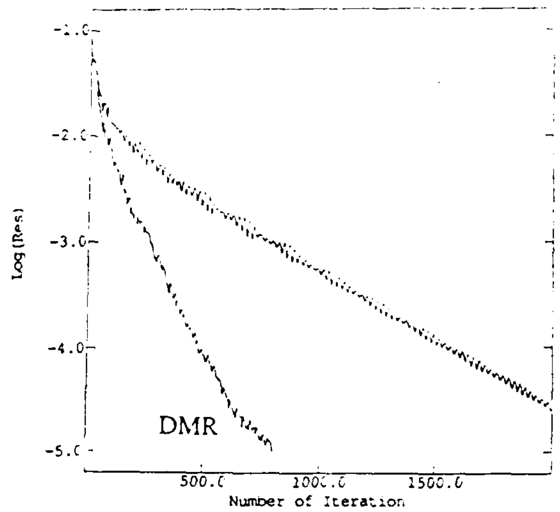


Figure 2. Comparison of convergence rates in terms of iteration numbers: non-accelerated and DMR accelerated algorithm for Ni's bump with $M_\infty = 0.5$.

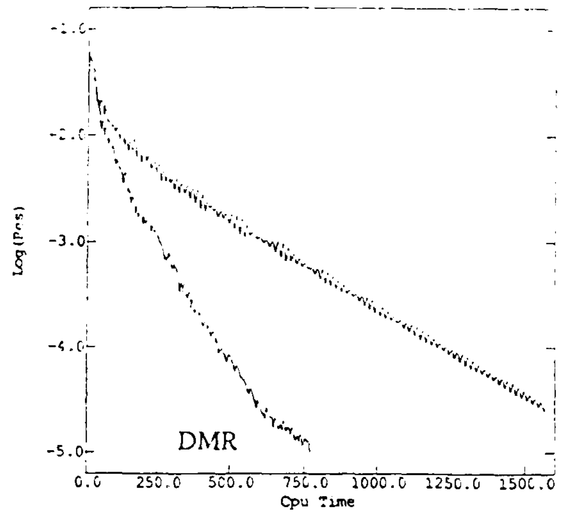


Figure 3. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for Ni's bump with $M_\infty = 0.5$.

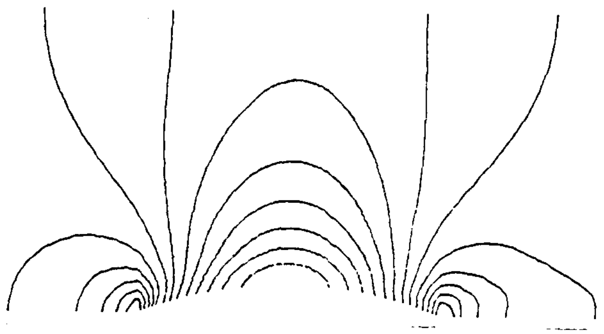


Figure 4. Constant pressure contours for non-accelerated algorithm for Ni's bump with $M_\infty = 0.5$.

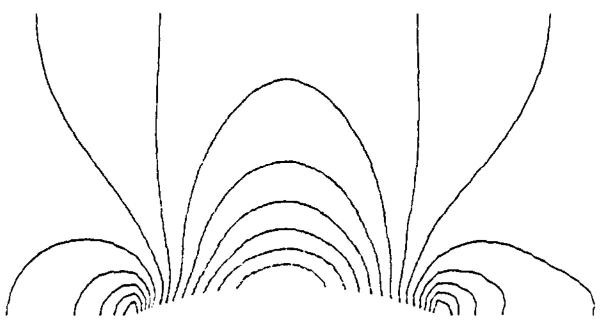


Figure 5. Constant pressure contours for DMR accelerated algorithm for Ni's bump with $M_\infty = 0.5$.

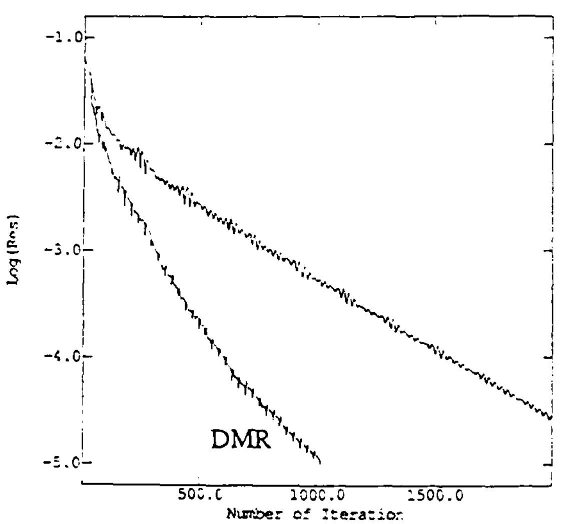


Figure 6. Comparison of convergence rates in terms of interation numbers: non-accelerated and DMR accelerated algorithm for Ni's bump with $M_\infty = 0.55$.

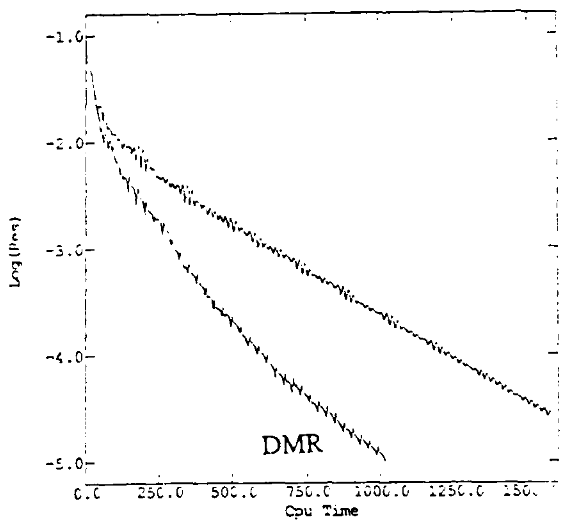


Figure 7. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for Euler Ni's bump with $M_\infty = 0.55$.

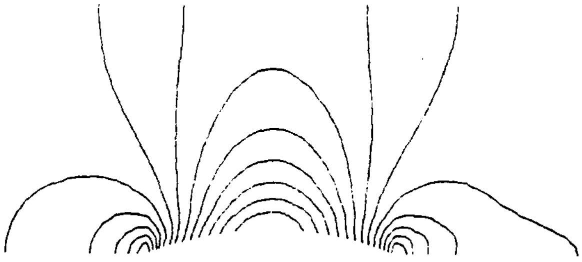


Figure 8. Constant pressure contours for non-accelerated algorithm for Ni's bump with $M_\infty = 0.55$.

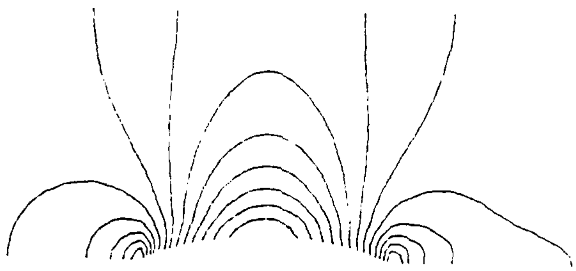


Figure 9. Constant pressure contours for DMR accelerated algorithm for Ni's bump with $M_\infty = 0.55$.

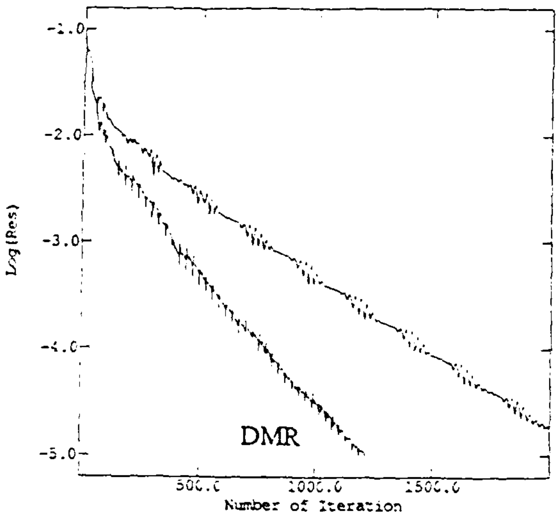


Figure 10. Comparison of convergence rates in terms of iteration numbers: non-accelerated and DMR accelerated algorithm for Ni's bump with $M_\infty = 0.6$.

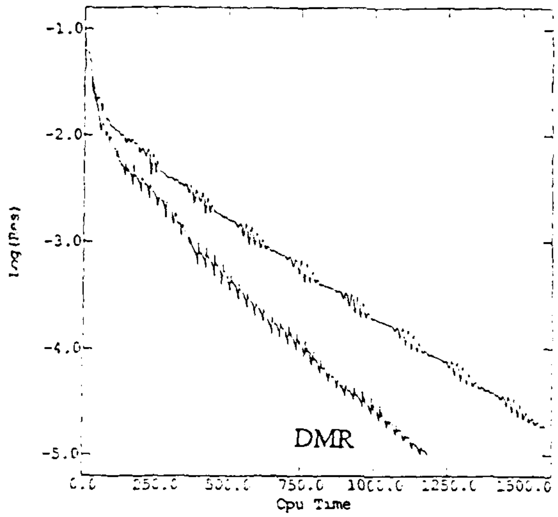


Figure 11. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for Ni's bump with $M_\infty = 0.6$.

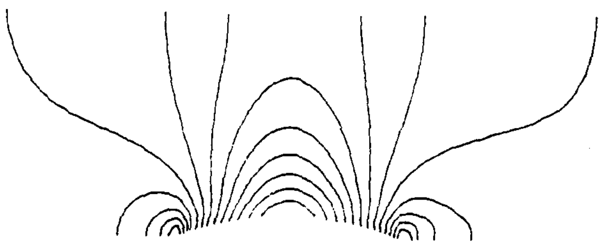


Figure 12. Constant pressure contours for non-accelerated algorithm for Ni's bump with $M_\infty = 0.6$.

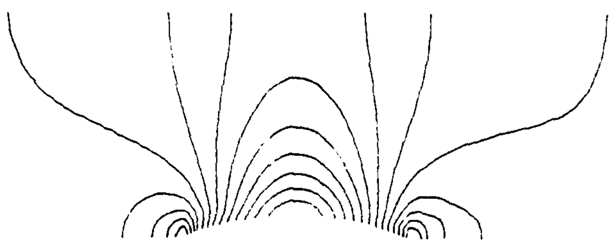


Figure 13. Constant pressure contours for DMR accelerated algorithm for Ni's bump with $M_\infty = 0.6$.

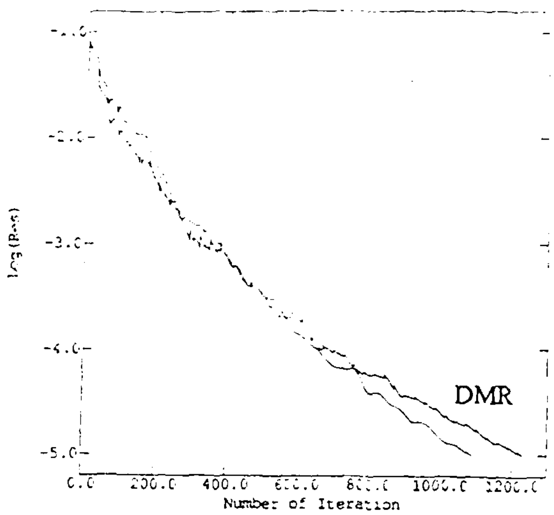


Figure 14. Comparison of convergence rates in terms of iteration numbers: non-accelerated and DMR accelerated algorithm for Ni's bump with M_∞ = 0.675.

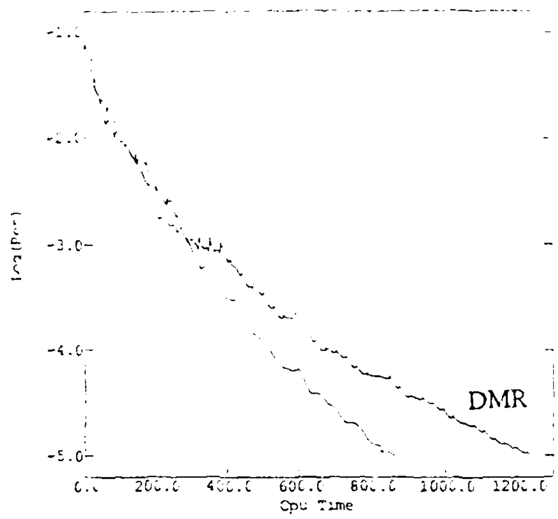


Figure 15. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for Ni's bump with M_∞ = 0.675.

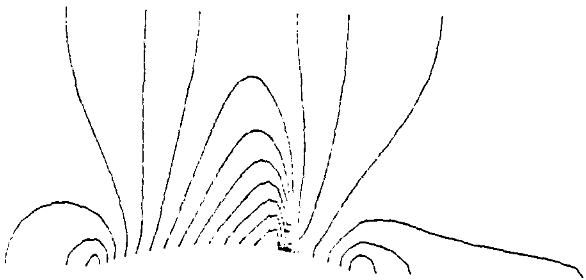


Figure 16. Constant pressure contours for non-accelerated algorithms for Ni's bump with M_∞ = 0.675.

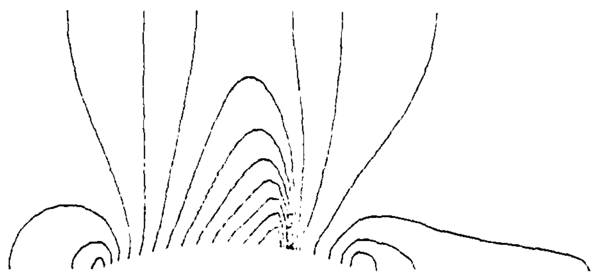


Figure 17. Constant pressure contours for non-accelerated algorithm for Ni's bump with M_∞ = 0.675.

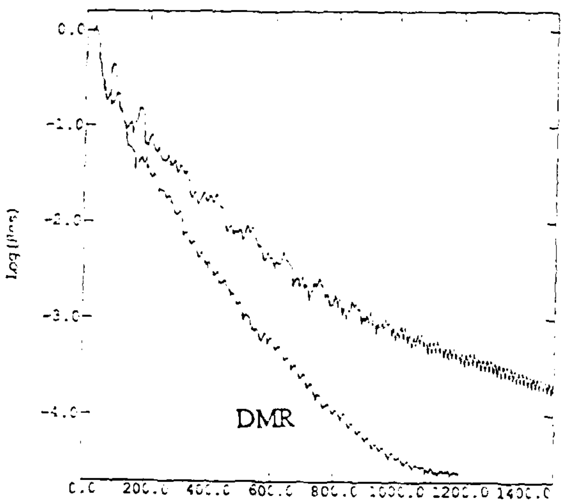


Figure 18. Comparison of convergence rates in terms of iteration numbers: non-accelerated and DMR accelerated algorithm for circle with M_∞ = 0.3.

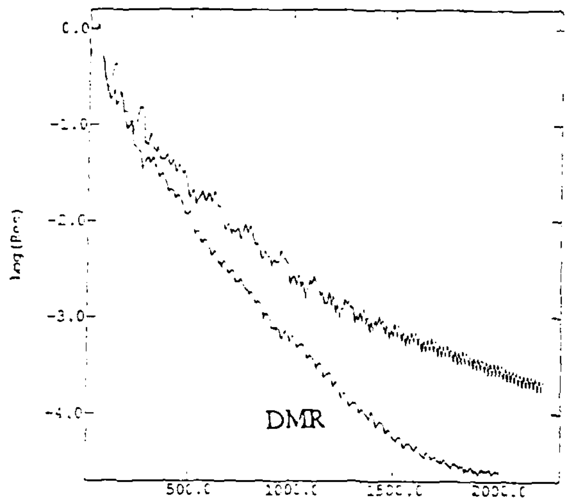


Figure 19. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for circle with M_∞ = 0.3.

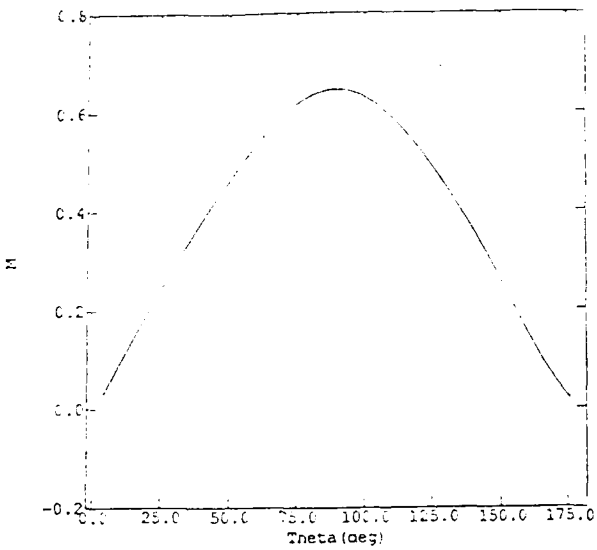


Figure 20. Surface Mach numbers for non-accelerated and DMR accelerated algorithm for circle with $M_\infty = 0.3$.

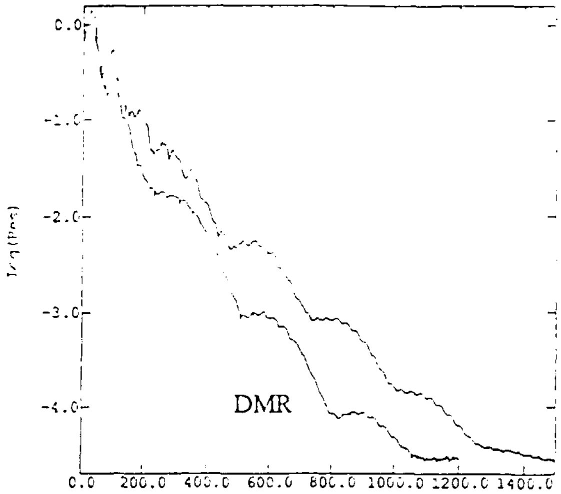


Figure 21. Comparison of convergence rates in terms of iteration numbers: non-accelerated and DMR accelerated algorithm for circle with $M_\infty = 0.4$.

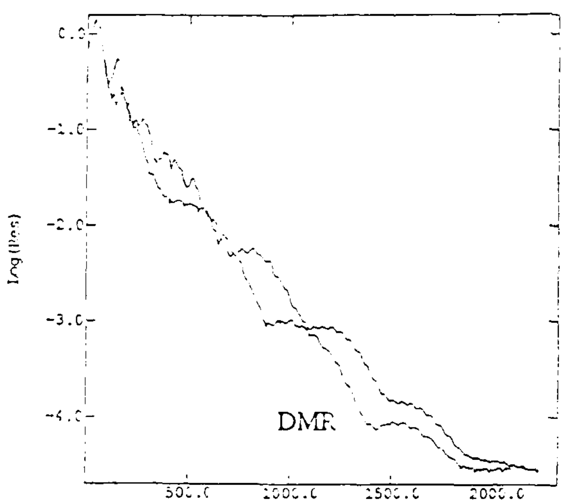


Figure 22. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for circle with $M_\infty = 0.4$.

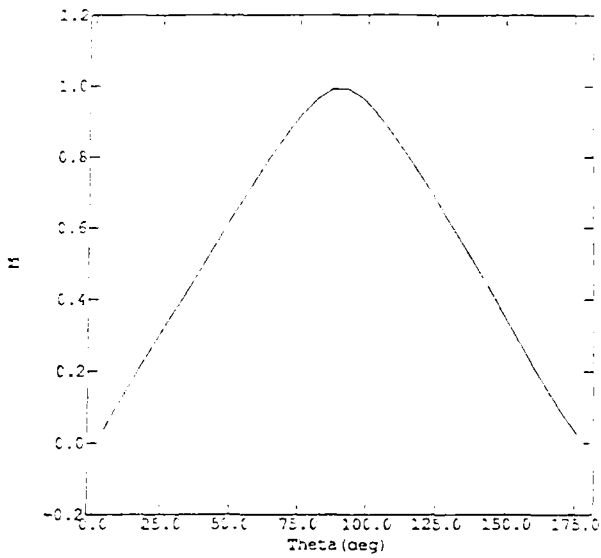


Figure 23. Surface Mach numbers for non-accelerated and DMR accelerated algorithm for circle with $M_\infty = 0.4$.

52

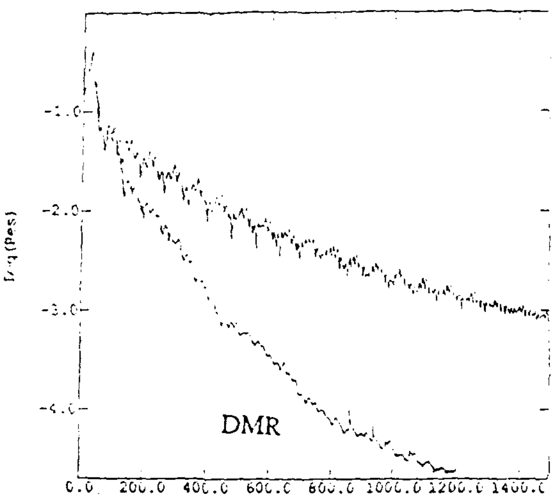


Figure 24. Comparison of convergence rates in terms of *iteration numbers*: non-accelerated and DMR accelerated algorithm for circle with $M_\infty = 0.1$.

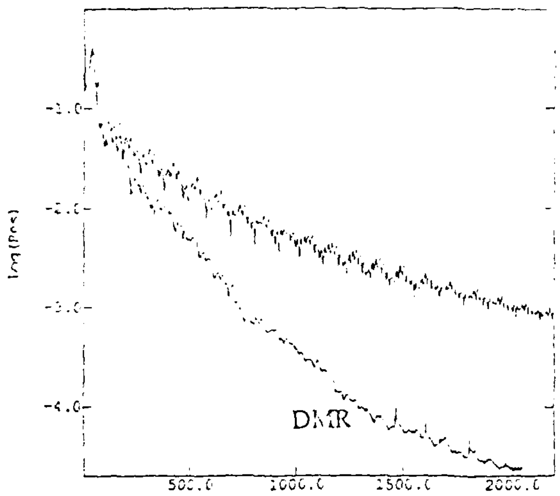


Figure 25. Comparison of convergence rates in terms of the CPU time: non-accelerated and DMR accelerated algorithm for circle with $M_\infty = 0.1$.

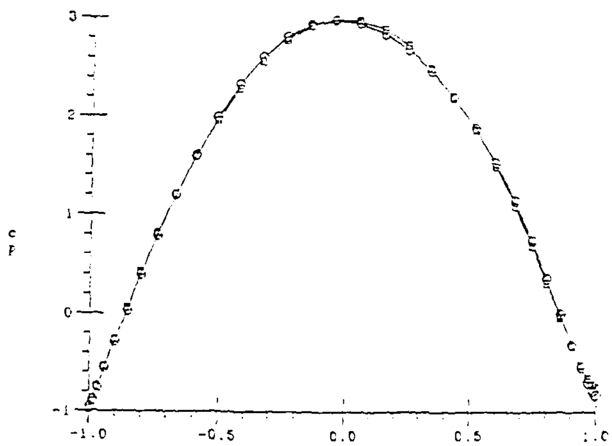


Figure 26. Comparison of surface pressure coefficients: non-accelerated (—□—□—□—) and DMR (—c—c—c—c—) accelerated algorithm for circle with $M_\infty = 0.1$.

ACCELERATION OF ITERATIVE ALGORITHMS

FOR EULER EQUATIONS OF GASDYNAMICS

Seungsoo Lee* and George S. Dulikravich**

Department of Aerospace Engineering

Pennsylvania State University, University Park, PA 16802

*Graduate Student.

**Associate Professor, Senior Member AIAA.

This paper was presented as AIAA paper 89-0097 at the AIAA Aerospace Sciences Meeting, Reno, NV, Jan. 9-12, 1989.

57

INTRODUCTION

One of the successful, explicit methods used to solve Euler and Navier-Stokes equations governing compressible flows is the finite volume Runge-Kutta time-stepping algorithm [1]. Several attempts have been made to accelerate the iterative convergence of this method. These acceleration methods are based on local time stepping [1], implicit residual smoothing [1], enthalpy damping [1] and multigrid techniques [2]. Also, an extrapolation procedure based on the power method and the Minimal Residual Method (MRM) were applied [2] to the Jameson's multigrid algorithm. The MRM has not been shown to accelerate the scheme without multigridding. It uses same values of optimal weights for the corrections to every equation in a system. If each component of the solution vector in a system of equations is allowed to have its own convergence speed, then a separate sequence of optimal weights could be assigned to each equation. This idea is the essence of the Distributed Minimal Residual (DMR) method [3] which is based on the General Nonlinear Minimal Residual (GNLMR) concept [4].

TIME-DEPENDENT EULER EQUATIONS

The system of time-dependent Euler equations of gasdynamics in a two-dimensional space can be written in a general conservative form [1] as

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial \xi} + \frac{\partial F}{\partial \eta} = 0 \qquad (1)$$

where the global solution vectors combining mass, ξ-momentum, η-momentum and energy conservation equations are defined as

$$Q = \frac{1}{D}\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_o \end{bmatrix} \qquad E = \frac{1}{D}\begin{bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ \rho(e_o + p)U \end{bmatrix} \qquad F = \frac{1}{D}\begin{bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ \rho(e_o + p)V \end{bmatrix} \qquad (2)$$

Here, t, ρ, u, v, p, e_o are time, density, x and y components of the velocity vector, thermodynamic pressure, and mass-specific total energy, respectively. In addition, U, V, ξ, η and D are the contravariant velocity vector components, non-orthogonal curvilinear computational coordinates, and determinant of the Jacobian of the transformation, $\partial(\xi,\eta)/\partial(x,y)$, respectively.

DISTRIBUTED MINIMAL RESIDUAL (DMR) METHOD

Local residual of the fini· -volume method at the global time level t can be expressed as

$$r^t = \iint \frac{\partial Q^t}{\partial t} dS = -\iint \left(\frac{\partial E^t}{\partial \xi} + \frac{\partial F^t}{\partial \eta}\right) dS \tag{3}$$

where S is the surface of the single grid cell and the components Q^t, E^t and F^t are defined in Eq. 2. In the DMR, corrections from M consecutive time levels are used to update the value of Q to (t+1) global time level. Thus,

$$Q^{t+1} = Q^t + \sum_m^M \Omega^m \tag{4}$$

where

$$\Omega^m = \begin{bmatrix} \omega_1^m & \Delta_1^m \\ \vdots & \vdots \\ \omega_L^m & \Delta_L^m \end{bmatrix} \tag{5}$$

Here, Δ_l^m are the corrections computed with the basic algorithm and ω_l^m are the weights for each of the l=1....L equations in the system (Eq. 2) at each of the m=1,..., M consecutive time levels. Therefore, the new local residual for the single grid cell will be

$$r^{t+1} = -\iint \left(\frac{\partial E^{t+1}}{\partial \xi} + \frac{\partial F^{t+1}}{\partial \eta}\right) dS \tag{6}$$

Using a Taylor series expansion of E^{t+1} and F^{t+1} in time and truncating it after the first term results in

$$r^{t+1} = r^t - \sum_m^M \iint \left(\frac{\partial}{\partial \xi}\left(\frac{\partial E^t}{\partial Q^t}\Omega^m\right) + \frac{\partial}{\partial \eta}\left(\frac{\partial F^t}{\partial Q^t}\Omega^m\right)\right) dS \tag{7}$$

Define the global residual R^{t+1} at the global time level (t+1) as a sum of the squares of the local residuals, that is,

$$R^{t+1} = \sum_{i}^{I} \sum_{j}^{J} (r^{t+1})^{*} (r^{t+1}) \tag{8}$$

where I and J define the grid size and the superscript * designates the transpose. The objective is to find optimum values of L sequences of M values of u_l^m that will minimize the global residual R^{t+1} at the next global time level (t+1). To minimize R^{t+1}, it is necessary to use the values u_l^m that satisfy

$$\frac{\partial R^{t+1}}{\partial u_l^m} = 0 \tag{9}$$

for all m and l. Thus, from Eq. 7, 8 and 9 it follows that

$$\sum_{i}^{I} \sum_{j}^{J} \{r^t - \sum_{n}^{M} \iint [\frac{\partial}{\partial \xi} (\frac{\partial E^t}{\partial Q^t} \Omega^n) + \frac{\partial}{\partial \eta} (\frac{\partial F^t}{\partial Q^t} \Omega^n)] \, dS\}^{*}$$

$$\cdot \{\iint [\frac{\partial}{\partial \xi} (\frac{\partial E^t}{\partial Q^t} \frac{\partial \Omega^m}{\partial u_l^m}) + \frac{\partial}{\partial \eta} (\frac{\partial F^t}{\partial Q^t} \frac{\partial \Omega^m}{\partial u_l^m})] \, dS\} = 0 \tag{10}$$

where

$$\frac{\partial \Omega^m}{\partial u_l^m} = \{\Delta_k^m \delta_{kl}\} \tag{11}$$

and δ_{kl} is the Kronecker delta. Notice that

$$\Omega^n = \sum_{q}^{L} \frac{\partial \Omega^n}{\partial u_q^n} u_q^n \tag{12}$$

Let

$$A_l^m = \iint [\frac{\partial}{\partial \xi} (\frac{\partial E^t}{\partial Q^t} \frac{\partial \Omega^m}{\partial u_l^m}) + \frac{\partial}{\partial \eta} (\frac{\partial F^t}{\partial Q^t} \frac{\partial \Omega^m}{\partial u_l^m})] \, ds \tag{13}$$

Note that A_l^m is not a function of u's. Then, Eq. (10) becomes

$$\sum_{i}^{I}\sum_{j}^{J}\sum_{n}^{M}\sum_{q}^{L} (A_q^n)^* A_i^m u_q^n = \sum_{i}^{I}\sum_{j}^{J} (r^t)^* A_i^m \tag{14}$$

Let

$$C_{ql}^{nm} = \sum_{i}^{I}\sum_{j}^{J} (A_q^n)^* A_l^m \quad B_l^m = \sum_{i}^{I}\sum_{j}^{J} (r^t)^* A_l^m \tag{15}$$

The result is a system of LxM equations

$$\sum_{n}^{M} (u_1^n C_{1l}^{nm} + u_2^n C_{2l}^{nm} + u_3^n C_{3l}^{nm} + \ldots\ldots + u_L^n C_{Ll}^{nm}) = B_l^m \tag{16}$$

for the LxM unknown optimal acceleration factors u_l^m. The DMR applied to the finite volume scheme [1] in two-dimensional case needs approximately 150% increase and in the three-dimensional case it needs approximately 175% increase in computer memory over the original non-accelerated algorithm [1]. Boundary conditions on the residuals in the integrals of Eq. 13 used extrapolation of the residuals from the interior of the flowfield.

RESULTS

All computations were performed on a VAX 11/8550 computer in a single precision mode. The first sequence of tests was performed on the internal two-dimensional (L = 4) flow problems by combining four consecutive time steps (M = 4). This means that a 16x16 matrix (Eq. 16) needs to be inverted. The test geometry was a 10% thick circular half airfoil on a wall of a straight two dimensional channel. The H-type grid size was 65x17 points. The calculations were started with uniform flow and the DMR was applied once after every 30 steps performed with the original unaccelerated code [1]. Figures 1 and 2 depict the convergence histories of flow calculations with $M_\infty = 0.5$ and $M_\infty = 0.675$. For the entirely subsonic flow ($M_\infty = 0.5$) the number of iterations needed to achieve the same level of residual is reduced almost by 60%, while the saving in

computational time is about 50%. Both figures indicate that DMR in its present version does not accelerate transonic flow (M_∞ = 0.675) computations. Superimposed constant pressure contours (Fig. 3) of the entire flow field for both the non-accelerated and the DMR accelerated schemes confirm that DMR method does not adversely influence the quality of the solution.

The second test case was a flow around a circle. An O-type radially clustered grid consisting of 64x32 grid cells was used. We applied DMR after every 60 iterations by combining four consecutive time levels. When the critical free stream (M_∞ = 0.4) was used, Figs. 4 and 5 indicate that the DMR method in its present form offers practically no gain. At very low free stream Mach numbers the system of Euler equations becomes very stiff, thus rapidly reducing the convergence rate of the non-accelerated scheme. On the other hand, when using M_∞ = 0.1, the DMR offers over 70% savings in the CPU time (Fig. 5) over the non-accelerated scheme.

Notice that all numerical results were obtained without the standard acceleration techniques such as explicit and implicit residual smoothing, enthalpy damping, multigridding and vectorization. These methods could be added to further accelerate the algorithm. The method seems to offer substantial time savings when applied to compressible flow codes at low Mach numbers.

CONCLUSIONS

A new method for the acceleration of explicit iterative algorithms for the numerical solution of systems of partial differential equations has been developed. The method is based on the idea of allowing each partial differential equation in the system to approach the converged solution at its own optimal speed while at the same time communicating with the rest of the equations in the system. The DMR *(Distributed Minimal Residual)* method computes a separate sequence of optimal acceleration factors to be used for each

component of the general solution vector. The acceleration scheme was applied to the system of time-dependent Euler equations of inviscid gasdynamics in conjunction with the finite volume Runge-Kutta explicit time-stepping algorithm. Using DMR without multigridding, between 30% and 70% of the total computational efforts were saved in the subsonic compressible flow calculations. The DMR method seems to be especially suitable for stiff systems of equations and can be applied to other systems of differential equations and other numerical algorithms.

ACKNOWLEDGEMENTS

This work was supported by the Air Force Office of Scientific Research/Numerical Mathematics Program under the grant AFOSR 87-0121. The authors are grateful for the expert typing by Ms. Amy Myers.

REFERENCES

1. Jameson, A., Schmidt, W. and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper No. 81-1259, Palo Alto, CA, June, 1981.

2. Hafez, M., Parlette, E., and Salas, M. D., "Convergence Acceleration of Iterative Solutions of Euler Equtions for Transonic Flow Computations," AIAA Paper 85-1641, July 1985.

3. Dulikravich, G. S., Dorney, D. J. and Lee, S., "Iterative Acceleration and Physically Based Dissipation for Euler Equations of Gasdynamics," ASME WAM '88, Symposia on Advances and Applications in Computational Fluid Dynamics, Chicago, IL, Oct. 28-Dec. 2, 1988.

4. Huang, C. Y., and Dulikravich, G. S., "Fast Iterative Algorithms Based on Optimized Explicit Time-Stepping," Computer Methods in Applied Mechanics and Engineering, 63, August 1987, pp. 15-36.

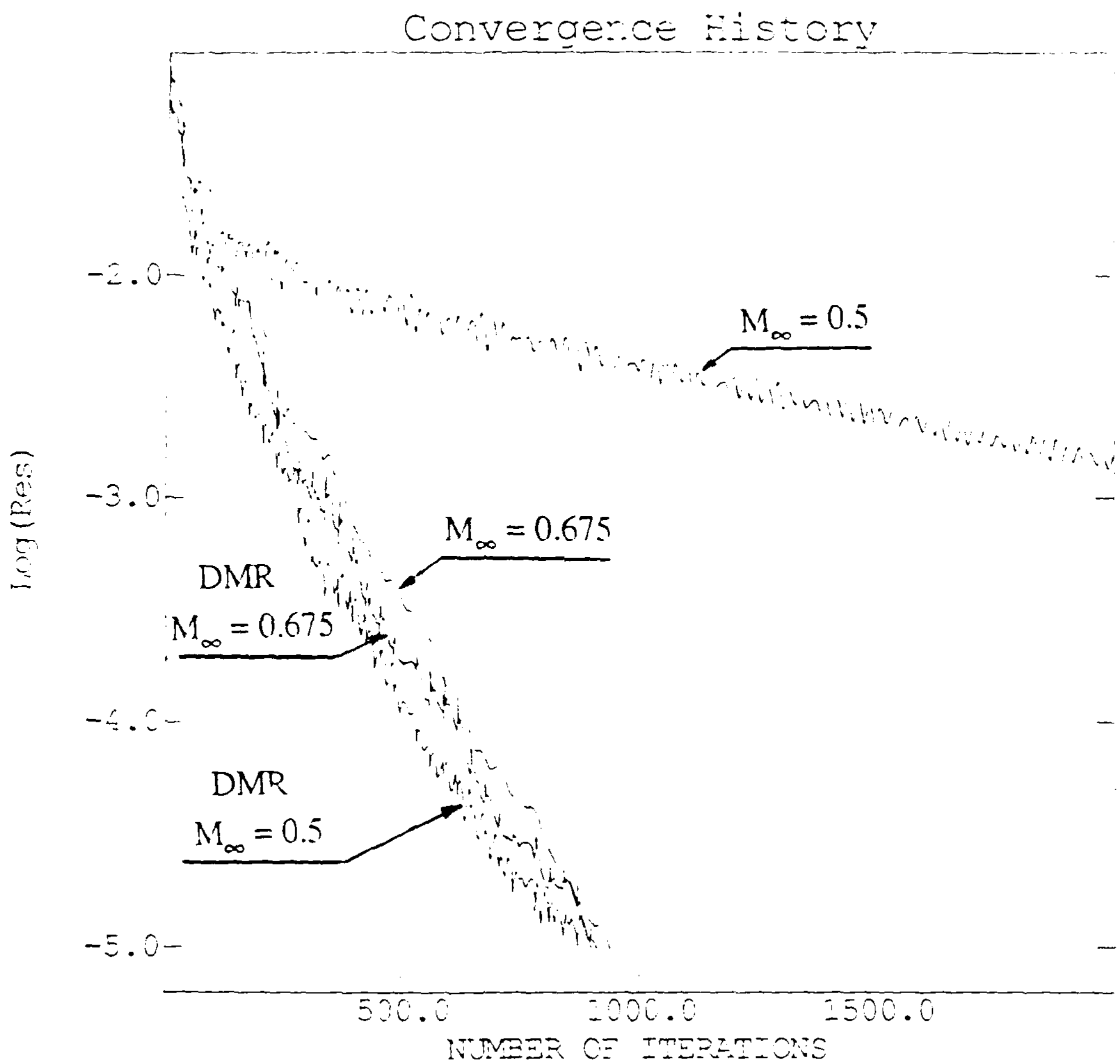
Figure 1. Convergence histories in terms of iteration numbers: non-accelerated (--------)and DMR accelerated (————) algorithm: channel flow.

Figure 2. Convergence histories in terms of the CPU time: non-accelerated (-----) and DMR accelerated algorithm: channel flow.

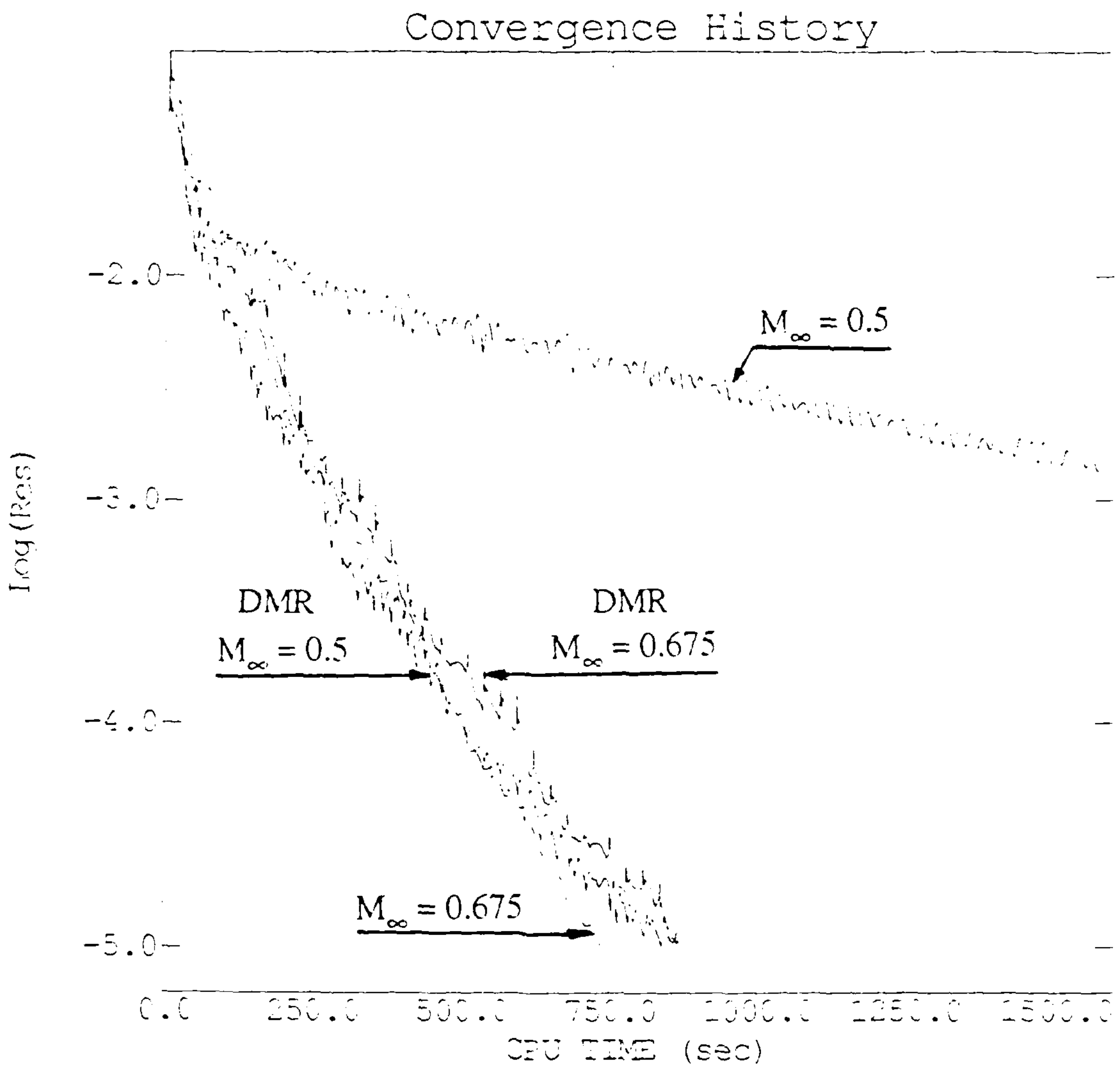
Figure 3. Constant pressure contours for DMR accelerated algorithm: transonic channel flow with $M_\infty = 0.675$.

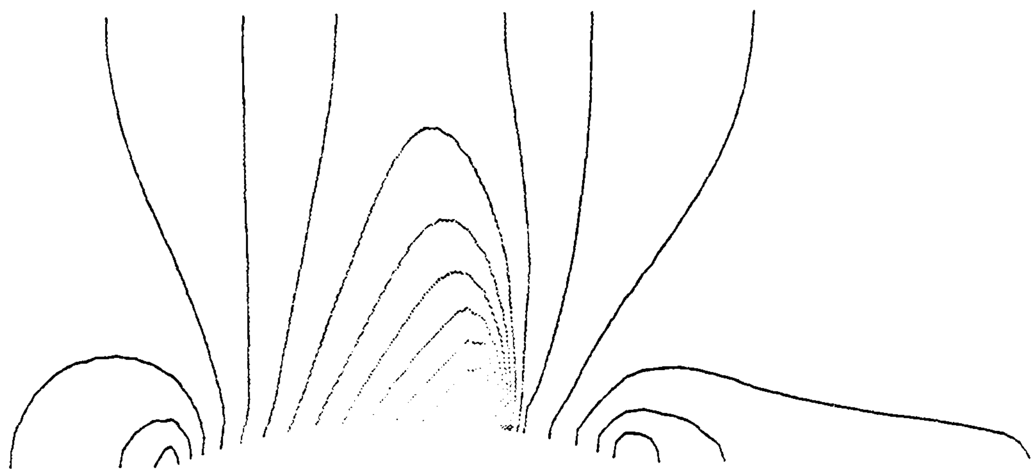
Figure 4. Convergence histories in terms of iteration numbers: non-accelerated (------) and DMR accelerated (———) algorithm: circle flow

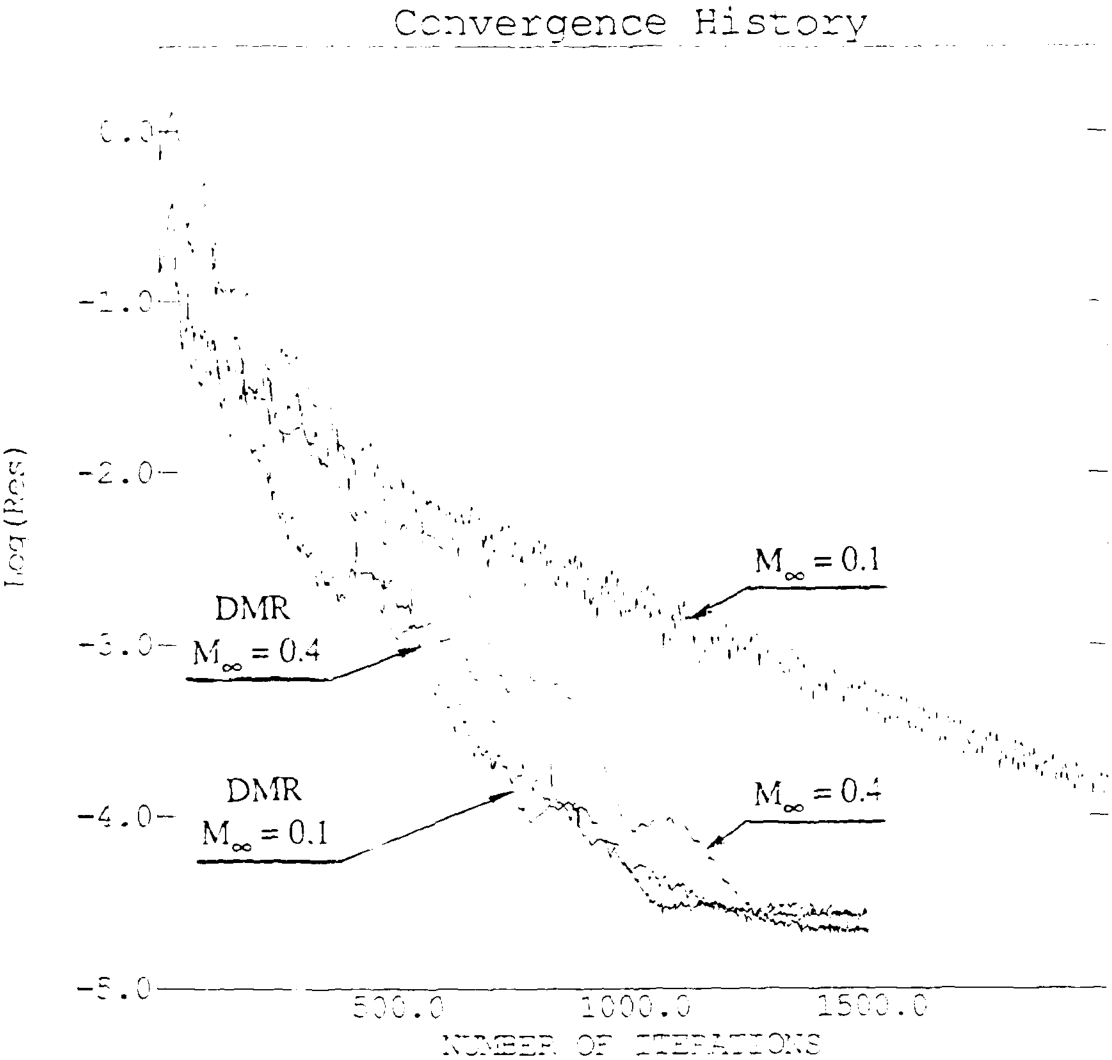
Figure 5. Convergence histories in terms of the CPU time: non-accelerated (-----) and DMR accelerated (————) algorithm: circle flow

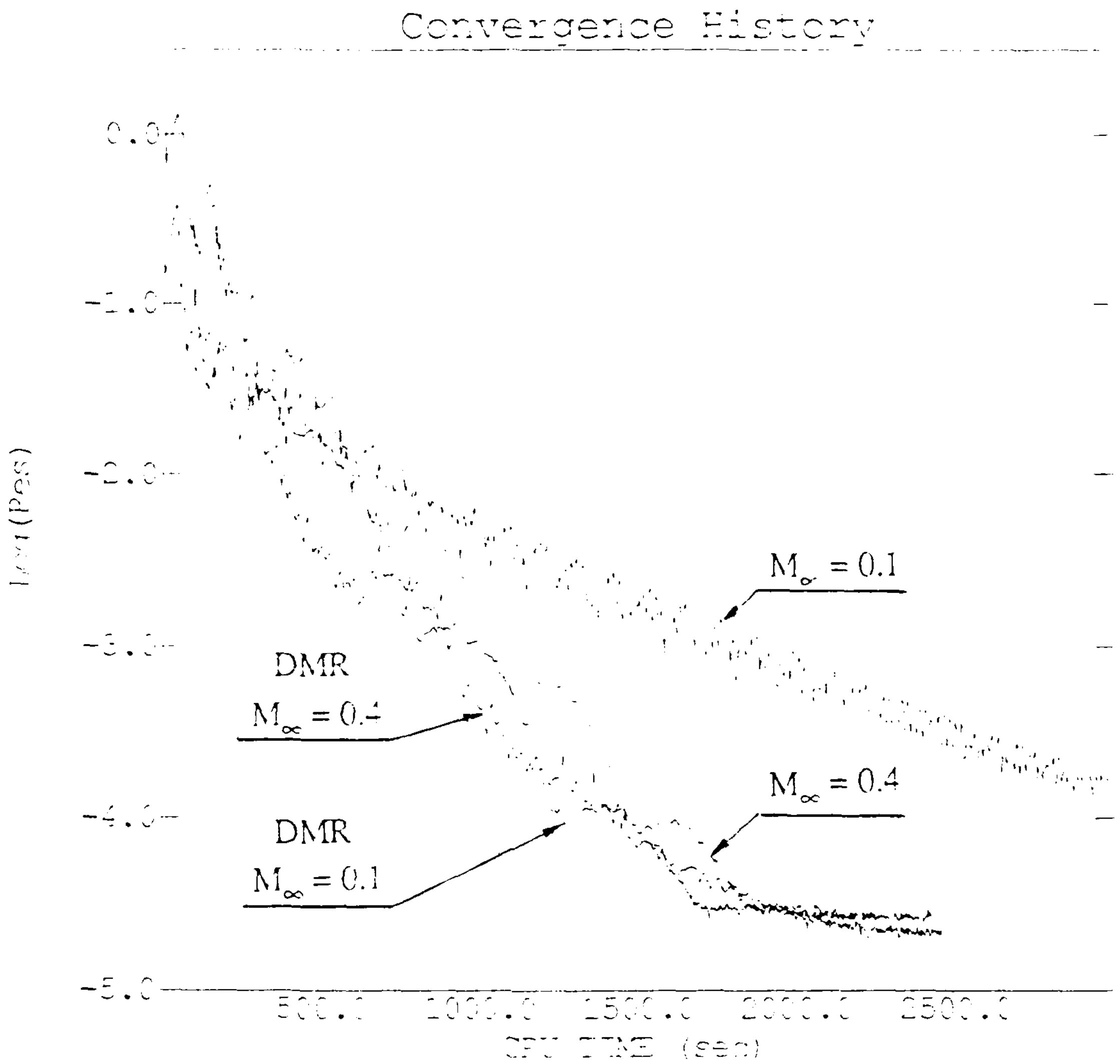

Convergence History
-2.0
-3.0
-4.0
-5.0
Log (Res)
$M_\infty = 0.5$
$M_\infty = 0.675$
DMR
$M_\infty = 0.675$
DMR
$M_\infty = 0.5$
500.0
1000.0
1500.0
NUMBER OF ITERATIONS

62


Convergence History
-2.0
-3.0
-4.0
-5.0
Log (Res)
$M_\infty = 0.5$
DMR
$M_\infty = 0.5$
DMR
$M_\infty = 0.675$
$M_\infty = 0.675$
0.0
250.0
500.0
750.0
1000.0
1250.0
1500.0
CPU TIME (sec)




Convergence History
Log (Res)
0.0
-1.0
-2.0
-3.0
-4.0
-5.0
500.0
1000.0
1500.0
NUMBER OF ITERATIONS
DMR
M∞ = 0.4
DMR
M∞ = 0.1
M∞ = 0.1
M∞ = 0.4



Convergence History
0.0
-1.0
-2.0
-3.0
-4.0
-5.0
500.0
1000.0
1500.0
2000.0
2500.0
CPU TIME (sec)
$M_\infty = 0.1$
DMR
$M_\infty = 0.4$
$M_\infty = 0.4$
DMR
$M_\infty = 0.1$